A Survey on Interaction Design with Large Language Models for Ontology Requirements Elicitation with Competency Questions

Yihang Zhao [a], Xi Hu [a], Timothy Neate [a], Albert Meroño Peñuela [a] and Elena Simperl [a,b]
[a] *Department of Informatics, King's College London, London, United Kingdom*
E-mails: yihang.zhao@kcl.ac.uk, xi.hu@kcl.ac.uk, timothy.neate@kcl.ac.uk, albert.merono@kcl.ac.uk, elena.simperl@kcl.ac.uk
[b] *Technical University of Munich, Munich, Germany*

Abstract. Ontology Engineering (OE) typically begins with ontology requirements elicitation, in which ontology engineers define the scope of concepts and relations an ontology must cover to adequately serve its intended application domain. Large Language Models (LLMs) can generate a large set of Competency Question (CQ) candidates to support this process; however, a target ontology scope is often multi-dimensional, ill-defined, and cannot be fully anticipated in advance, and existing tools provide little support for ontology engineers to subsequently explore and generate new CQ candidates (divergent thinking), evaluate, refine, and eliminate existing ones (convergent thinking), and thus progressively define a well-scoped ontology. We argue that interaction designs from LLM-based systems in arts and creativity domains, where divergent and convergent thinking have been extensively studied, are transferable to ontology requirements elicitation. To identify these designs, we conducted a Systematic Literature Review (SLR) of 50 papers, identifying 7 Interaction Techniques (ITs) and 14 User Interfaces (UIs), each justified with respect to how it supports divergent thinking, convergent thinking, or both. To explore the transferability and applicability of the identified ITs and UIs to LLM-based ontology scoping, we conducted a design thinking workshop (N=7) that produced a conceptual interaction model, a system prototype called OntoScope implementing that model, and a use case demonstration showing how OntoScope can potentially support an OE expert in scoping a university ontology. The identified ITs and UIs can serve as a reference for tool developers working on ontology requirements elicitation, broader OE tasks that require human reasoning and auditing over LLM-generated content, or designing user-friendly OE tools for domain experts and end users without prior OE expertise.

Keywords: ontology engineering, ontology requirements elicitation, competency questions, large language models, interaction design, user interface design, divergent thinking, convergent thinking, systematic literature review, human-computer interaction, ontology scoping, knowledge graph, semantic web, design thinking, ontoscope

1. Introduction

Ontology, originally a philosophical term, refers to the study of being [74]. In Information and Communication Technologies (ICT), an ontology can be understood as a formal representation of a domain as a set of concepts and the relations that hold between them [87, 115, 138]. Such representations can ensure a common understanding of information and make explicit domain assumptions, thus allowing organisations to make better sense of their data [64]. OE can be broadly understood as referring to the activities that concern the ontology development process, its life cycle, and the methodologies, tools, and languages for building, maintaining, and reusing ontologies

[60, 138]. OE typically begins with ontology requirements elicitation, in which ontology engineers define the scope of concepts and relations an ontology must cover to adequately serve its intended application domain [7, 46].

To define the scope of an ontology, ontology engineers are often required to consider at least two key dimensions: horizontally, what subdomains to cover, and vertically, at what level of term granularity [7, 46, 183, 185]. This scoping process often relies on CQs, which are functional requirements of an ontology expressed as natural language questions that the ontology must be able to answer [180, 185]. CQ patterns vary with the underlying methodology [63, 115], templates [11, 85, 169], archetypes [126], intended use [7], and level of automation [5, 125]. In this work, we adopt the three CQ patterns defined by RevOnt [40], which have been validated across 20 representative domains on Wikidata[1] (the largest collaboratively curated Knowledge Graph (KG), covering widely shared knowledge across many domains [156]). As illustrated in Fig. 1, each pattern differs in which element of a verbalised schema-level Resource Description Framework (RDF)[2] triple (an ontology representation written as `(subject, predicate/property, object)`) is treated as the answer.

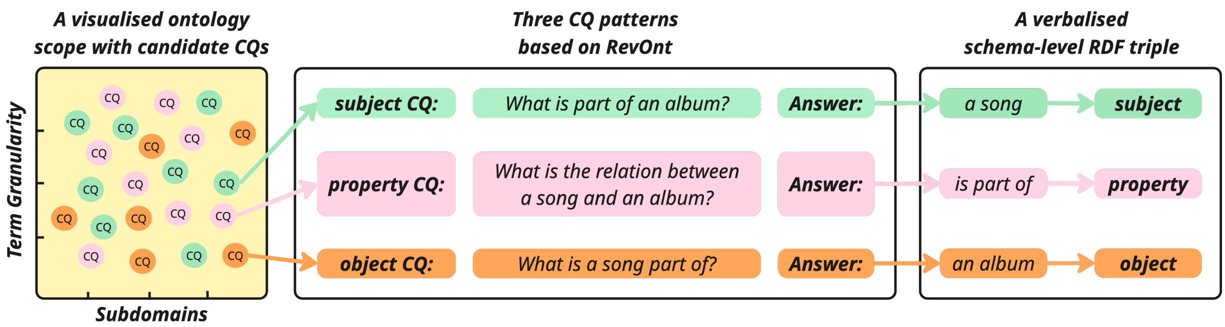


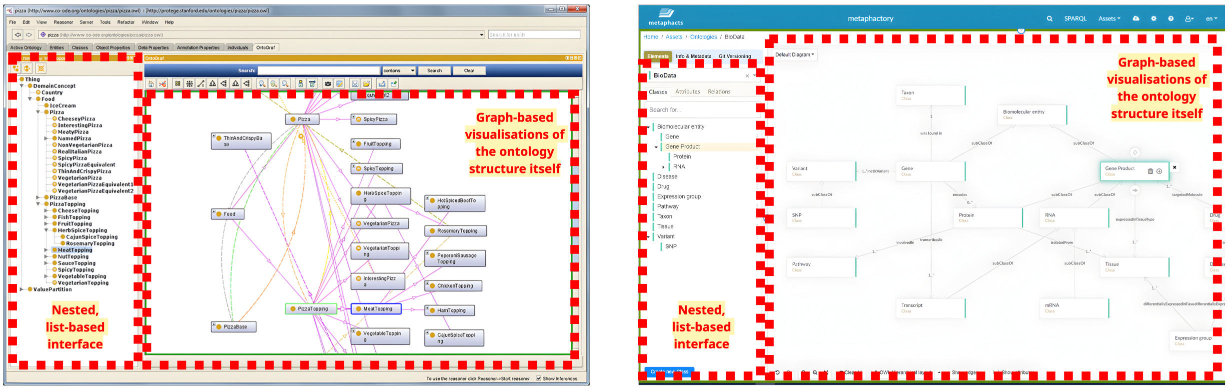
Fig. 1. Three CQ patterns based on RevOnt [40], each differing in which element of a verbalised schema-level RDF triple is treated as the answer: a subject CQ (e.g., "What is part of an album?", answer: *a song*), a property CQ (e.g., "What is the relation between a song and an album?", answer: *is part of*), and an object CQ (e.g., "What is a song part of?", answer: *an album*). The left panel shows a visualised ontology scope, where each dot represents a CQ together with its suggested term organised along two dimensions: subdomains (horizontal) and term (concept or relation) granularity (vertical).

Thanks to their generative capacities, current LLM-based systems can generate hundreds and thousands of CQ candidates to support this scoping process [5–8, 12, 40, 53, 180, 185]. Automated tools such as AgOCQs [12] and NeOn-GPT [53] can derive CQs from domain-specific text corpora, and RevOnt [40] and RETROFIT-CQs [5] can generate CQs from existing ontology vocabularies. However, these tools are designed for generation only and provide little support for ontology engineers to subsequently audit the generated candidates, leaving ontology engineers unsure whether the candidates are complete and relevant to define the target ontology scope [7, 46, 183, 185].

Existing tools that provide some scoping auditing support each have their own limitations. LLM-based conversational agents such as OntoChat [180, 185], which elicits ontology user stories from domain experts and converts them into CQs, support auditing through follow-up questions and varied suggestions, but its chatbot UI is inherently linear and its ITs are turn-based, potentially causing fixation [43, 50, 82, 145, 178] where ontology engineers may become anchored to their initial inputs, potentially leading to CQs that may not adequately reflect the intended ontology scope [7, 46, 183, 185]. Ontology editors such as Protégé [109] and metaphactory [69] provide two UIs: nested, list-based UIs that organise concepts and relations hierarchically, requiring constant scrolling to navigate the full scope and making it difficult to maintain a global overview; and graph-based UIs of the ontology structure itself, which can provide a global overview but are not organised along the dimensions of subdomains and granularity, making it difficult to reason about whether the CQ candidates collectively achieve the intended scope (Fig. 2a,

[1]https://www.wikidata.org/wiki/Wikidata:Main_Page
[2]https://www.w3.org/TR/1999/REC-rdf-syntax-19990222/

Fig. 2b). Additionally, general mind-mapping tools such as Miro[3] and XMind[4] offer flexible UIs for personalising the scope auditing process, but these UIs are underexplored for ontology scoping purposes.



(a) Protégé [109]. Screenshots taken from the tool, annotated for clarity.

(b) metaphactory [69]. Screenshots taken from the tool, annotated for clarity.

Fig. 2. Two representative ontology tools, each providing a nested, list-based interface (left panels) and a graph-based visualisation of the ontology structure (right panels). Neither interface type is organised along the dimensions of subdomains and granularity needed for systematic CQ-based scope auditing.

We argue these challenges can be addressed by drawing on interaction designs (ITs and UIs) of LLM-based systems that have been widely used in the arts and creativity domain for supporting divergent and convergent thinking [144, 145, 183]. As illustrated in Fig. 3, divergent thinking refers to a thought process in which users explore and generate many candidates along multiple dimensions without committing to any single one, with the goal of expanding the solution space [65]. Convergent thinking, by contrast, refers to a thought process in which users evaluate, refine, and eliminate candidates along the same dimensions by following logical and structured steps toward a defined goal, with the aim of narrowing the solution space [45, 65]. In the arts and creativity domain, users work toward a defined goal whose full shape cannot be anticipated in advance: the target solution space is multi-dimensional and ill-defined, requiring users to first expand it through divergent thinking and then narrow it through convergent thinking. To support this process, these systems often present candidates in a visualised space organised along key dimensions, allowing users to reason across dimensions and decide where to diverge or converge. While the number of dimensions can vary, two-dimensional spaces are most commonly adopted for manageability (e.g., [18, 23, 165, 167]). For example, in creative writing, these two dimensions may be style and tone; in visual design, they may be colour and composition.

Ontology scoping shares the same situation: the target ontology scope is multi-dimensional, ill-defined, and cannot be fully anticipated in advance. Ontology engineers must consider at least two key dimensions: subdomains and term granularity, where each generated CQ candidate suggests concepts and relations at a particular subdomain and level of term granularity [7, 46, 183, 185]. These candidates can therefore be organised on a visualised ontology scope with two dimensions, where subdomains run horizontally and term granularity runs vertically, enabling ontology engineers to reason across dimensions: identifying subdomains or granularity levels with too few CQ candidates as gaps where divergent thinking is needed (requiring LLM-based systems to support ontology engineers in exploring and generating new CQ candidates), and identifying those with redundant or conflicting candidates as overlaps where convergent thinking is needed (requiring LLM-based systems to support ontology engineers in refining and deleting existing CQ candidates), to progressively define a well-scoped ontology, as illustrated in Fig. 4.

To investigate what interaction designs of LLM-based systems in the arts and creativity domain can support ontology engineers in divergent and convergent thinking during LLM-based ontology scoping with CQs, and how these

[3]https://miro.com
[4]https://xmind.com

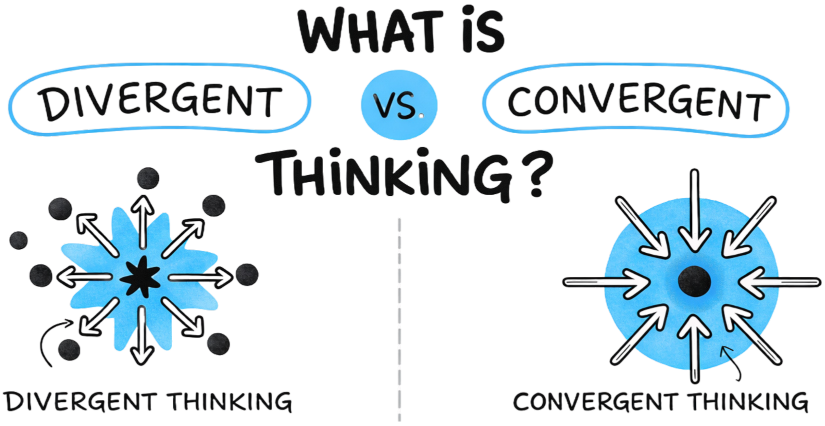


Fig. 3. Divergent thinking explores and generates many possible candidates along multiple dimensions without committing to any single one, with the goal of expanding the solution space, while convergent thinking evaluates, refines, and eliminates candidates along the same dimensions by following logical and structured steps toward a defined goal [65].

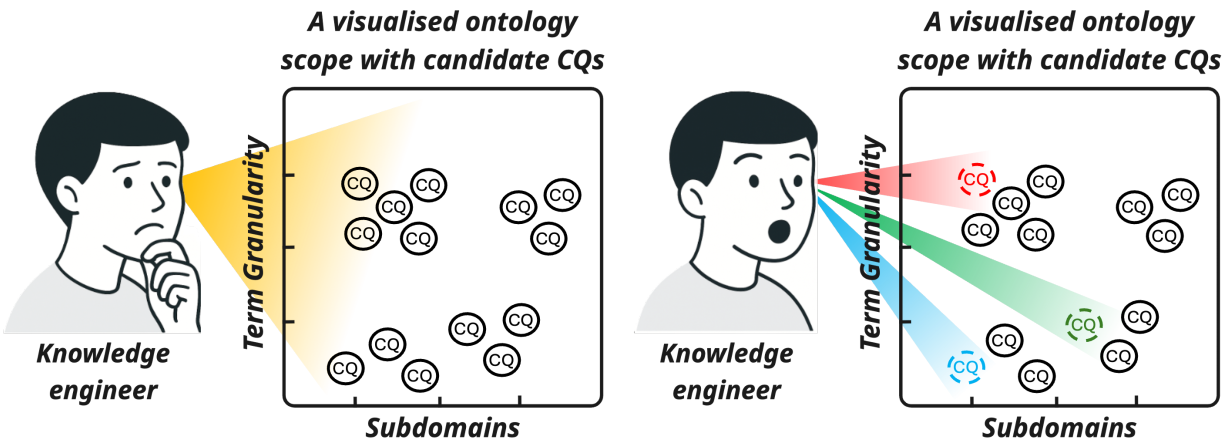


Fig. 4. Divergent and convergent thinking in ontology scoping. Left: an ontology engineer maintains a global overview of the candidate CQ space across the dimensions of subdomains (horizontal) and term granularity (vertical), enabling reasoning across dimensions. Right: based on this overview and dimensional reasoning, the ontology engineer performs divergent thinking by **adding** new candidate CQs and convergent thinking by **refining** or **deleting** existing CQs, to progressively define a well-scoped ontology.

designs can be translated into a concrete interaction model and evaluated in practice, we propose three Research Questions (RQs):

- **RQ1:** What ITs and UIs of LLM-based systems in the arts and creativity domain support divergent and convergent thinking?
- **RQ2:** How can the identified ITs and UIs be translated into a conceptual interaction model for LLM-based ontology scoping?
- **RQ3:** How applicable is the conceptual interaction model in supporting an ontology engineer in LLM-based ontology scoping in practice?

To address these RQs, we made the following contributions:

- A Systematic Literature Review (SLR)[5] of 50 papers identifying 7 ITs and 14 UIs from LLM-based systems

[5]https://github.com/King-s-Knowledge-Graph-Lab/OntoScope/blob/main/attached_assets/Codebook.xlsx

in arts and creativity domains, each justified with respect to how it supports divergent thinking, convergent thinking, or both, with Inter-Rater Reliability (IRR) reported per category (Section 3.5).
- A conceptual interaction model for LLM-based ontology scoping with CQs, produced through a design thinking workshop, that maps a selected subset of the identified ITs and UIs onto the three stages of Shneiderman's visual information-seeking mantra [136] (Table 7, as illustrated in Fig. 11, Section 4.2).
- OntoScope[6], a system prototype implementing the conceptual interaction model (also publicly accessible online[7]), and a use case demonstration showing how OntoScope can potentially support an ontology engineer in scoping a university ontology through divergent and convergent thinking (Section 4.3, Section 4.4, Section 4.5, as depicted in Fig. 12 and Fig. 13).

These contributions have implications at three levels. For ontology requirements elicitation, the identified ITs and UIs provide a reference for tool developers building LLM-based systems that support ontology engineers in progressively defining a well-scoped ontology through divergent and convergent thinking. For OE more broadly, the findings are potentially applicable to other OE tasks that involve human reasoning and auditing over LLM-generated content, such as user story elicitation [180], KG population [119], and ontology alignment [52]. For the Semantic Web community, the pool of identified interaction designs provides a basis for designing tools for domain experts and end users without prior OE expertise, by selecting interaction designs that align with their existing mental models and working practices rather than requiring them to adapt to tool designed for trained ontology engineers, supporting the broader goal of making OE tools accessible to a wider range of users [46, 138], and thus broadening participation in ontology building toward a more diverse and inclusive community of contributors.

The remainder of this paper is structured as follows. Section 2 reviews related work. Section 3 describes the SLR methodology. Section 3.5 presents the SLR findings. Section 4 describes the design thinking workshop, including participants (Section 4.1), the first workshop session covering Empathise, Define, Ideate, and Prototype stages and the conceptual interaction model produced (Section 4.2), OntoScope and its implementation (Section 4.3), the second workshop session covering the Test stage (Section 4.4), and the use case demonstration (Section 4.5). Section 5 discusses the findings and their implications. Section 6 presents the limitations of this work. Section 7 concludes the paper.

2. Related work

This section reviews four areas of related work. Section 2.1 details what CQs are, the three CQ patterns adopted in this paper, and how CQs are categorised by intended use. Section 2.2 traces the progression of ontology requirements elicitation methods, identifying at each stage the limitations in supporting ontology engineers to audit CQ candidates across the dimensions of subdomains and term granularity. Section 2.3 defines divergent and convergent thinking, introduces the Double Diamond model, and establishes the structural connection between ontology scoping and the solution space of divergent and convergent thinking. Section 2.4 reviews interaction designs for supporting divergent and convergent thinking in LLM-based systems, introducing Shneiderman's Visual Information-Seeking Mantra as the organising framework, and identifying the gap that their application to ontology scoping has yet to be explored.

2.1. *Competency questions in ontology engineering*

CQs are natural language questions that an ontology must be able to answer, serving as functional requirements that define what the ontology should and should not cover [7, 63]. They are widely adopted across OE methodologies, including Ontology Development 101 [115], NeOn [143], On-To-Knowledge [141], SAMOD [120], eXtreme Design [123], Pay as you go [132], and LOT [122], and support ontology construction, verification, evaluation, and potential reuse with a focus on completeness and relevance [7]. CQ patterns vary with the underlying methodology [63, 115], templates [11, 85, 169], archetypes [126], intended use [7], and level of automation [5, 125].

[6] https://github.com/King-s-Knowledge-Graph-Lab/OntoScope
[7] https://ontoscope.digital/

For example, Ren et al. [126] define CQ archetypes as syntactic patterns to be filled by domain experts (e.g. "Which `[CE1] [OPE] [CE2]`?", where `[CE1]` and `[CE2]` are class expressions and `[OPE]` is an object property expression); Bezerra and Freitas [21] identify patterns instantiated with ontology vocabulary elements for automatic testing; Wisniewski et al. [169] proposed 106 distinct patterns from 234 CQs across five ontologies; and CLaRO [85] provides a template-based Controlled Natural Language (CNL) with 93 templates and 41 variants, later expanded in CLaRO v2 [11] to 120 main templates derived from 329 CQs.

Based on an analysis of existing CQ patterns, Alharbi et al. [7] categorise CQs into two main types. The first type, *scoping CQs*, helps define the domain by specifying what concepts and relations the ontology should cover; for example, the scoping CQ "What is part of an album?" (answer: `a song`) contributes to defining the subdomain of musical composition at a specific level of term granularity. In this paper, we adopt the three scoping CQ patterns proposed by RevOnt [40] (Fig. 1), grounded in verbalised schema-level RDF triples written as `([subject], [predicate/property], [object])`, each differing in which element is treated as the answer: a subject CQ ("What `[property] [object]`?", answer: `[subject]`), a property CQ ("What is the relation between `[subject]` and `[object]`?", answer: `[property]`), and an object CQ ("What `[subject] [property]`?", answer: `[object]`). These patterns are adopted in RevOnt as an LLM-based CQ generation system validated across 20 representative domains on Wikidata [156], providing a validated specification of what a CQ and its answer express.

The second type, *verified CQs*, can be directly transformed into SPARQL queries and serve as benchmarks for ontology testing and validation, supported by tools such as OWLunit[8] for unit testing and OOPS! [121] for detecting common ontology errors. Beyond these two types, other CQs may be syntactically or semantically incorrect, such as ambiguous questions (e.g. "Which devices can I see?") or factually incorrect questions (e.g. "Which vegetarian pizza contains ham?") [7, 85], or not yet supported by the ontology, which Zhang et al. [180] define as *adversarial CQs* for ontology testing. In this paper, we focus on scoping CQs, as the central challenge is supporting ontology engineers in auditing generated CQ candidates to progressively define a well-scoped ontology.

2.2. *Ontology requirements elicitation with CQs*

Ontology requirements elicitation refers to the process by which ontology engineers gather knowledge from domain experts or existing knowledge sources (e.g. domain-specific text corpora or ontology vocabularies) to define the scope of a target ontology [115, 122, 123, 143]. This scope is considered along at least two key dimensions: subdomains (horizontal) and term granularity (vertical). The defined ontology scope serves as a reference for ontology design, evaluation, maintenance, and reuse [87].

To gather knowledge from domain experts, ontology engineers often conduct unstructured interviews to elicit user stories, including expert personas, goals, and the gaps the target ontology aims to address [42, 44, 180]. These user stories are then used to construct CQs, as they help ontology engineers understand what the ontology needs to cover from the perspective of its intended users. However, such interviews tend to anchor on the ontology engineer's existing mental model [111], particularly at early stages when ontology engineers have a limited understanding of the domain. This limits opportunities for domain experts to explore broader user stories, restricting divergent thinking and leaving the resulting CQs narrowly scoped.

Once an initial understanding is acquired, ontology engineers typically shift toward structured elicitation methods to transform user stories into CQs by clarifying key concepts, their relations, and the distinctions between them. Common approaches include card sorting [133], triad analysis [128], and the twenty questions technique [86]. However, these methods follow a predominantly convergent path, building on predefined knowledge from initial interviews with limited opportunity for expanding the scope across subdomains and levels of term granularity. Additionally, they are time-intensive and may be impractical for large-scale projects.

In response, LLM-based systems such as OntoChat [180, 185] emulate the role of a knowledge elicitor by gathering and refining user stories through prompt-driven conversational workflows, which are automatically converted into CQs, and supporting subsequent auditing through follow-up questions and varied suggestions (Fig. 5). However,

[8] https://github.com/luigi-asprino/owl-unit

its *linear conversational UI* and *turn-based IT* potentially cause fixation [43, 50, 82, 145, 178] where ontology engineers may become anchored to their initial inputs, narrowing the explored scope prematurely, thus leading to CQ candidates that may not adequately cover the intended subdomains and levels of term granularity [144, 145, 185].

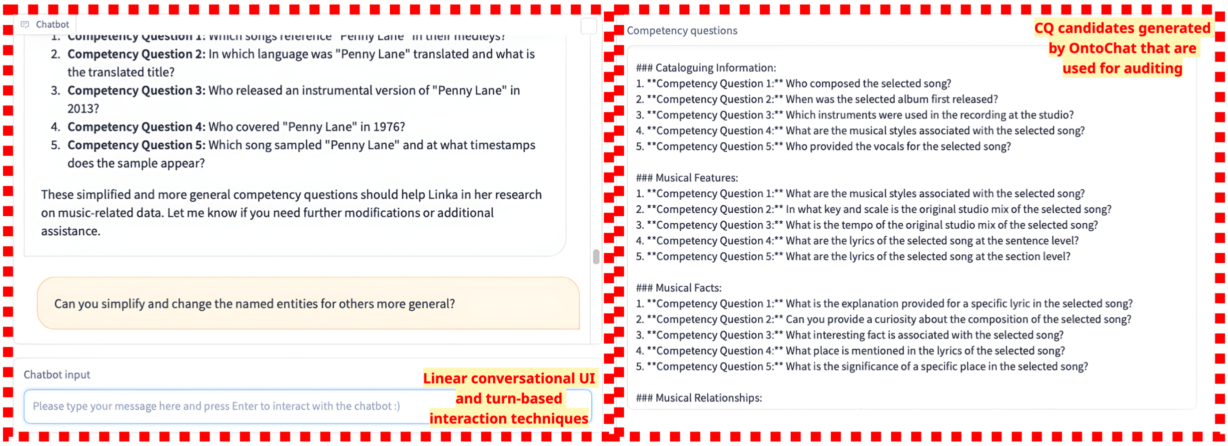


Fig. 5. The existing OntoChat system [180, 185]: a *linear conversational UI* and *turn-based ITs* (left panel) that generate CQ candidates used for auditing (right panel). Screenshot taken from the tool, annotated for clarity.

Another approach extracts knowledge from existing domain documentation rather than from domain experts directly. Traditionally, this involves manually authoring CQs [63, 115], often resulting in inconsistencies due to variations in authoring styles [7]. Semi-automated approaches using CNL [90], such as CLaRO [85] and others [11, 22, 126, 169], reduce this inconsistency through structured templates, but still require extensive manual effort for domain adaptation [7]. More recent LLM-based approaches generate CQs from text corpora (e.g. AgOCQs [12], NeOn-GPT [53]) or from existing ontology vocabularies (e.g. RevOnt [40], RETROFIT-CQs [5]), producing hundreds and thousands of CQ candidates, but are designed for generation only and provide little support for ontology engineers to subsequently audit them.

Ontology editors such as Protégé [109] and metaphactory [69] provide some auditing support through nested, list-based UIs and graph-based UIs of the ontology structure, but neither is organised along the dimensions of subdomains and granularity, making it difficult to reason about whether CQ candidates collectively achieve the intended scope. General mind-mapping tools such as Miro [3] and XMind[4] offer flexible UIs for personalising the scope auditing process, but these designs are underexplored for ontology scoping purposes. Thus, a key challenge remains: how to design ITs and UIs for LLM-based systems that support ontology engineers in reasoning across the dimensions of subdomains and term granularity to progressively audit and define a well-scoped ontology.

2.3. *Divergent and convergent thinking*

First coined by Guilford [65], divergent thinking refers to a thought process in which users explore and generate many possible candidates along multiple dimensions without committing to any single one, with the goal of expanding the solution space. Convergent thinking, by contrast, refers to a thought process in which users evaluate, refine, and eliminate candidates along the same dimensions by following logical and structured steps toward a defined goal, with the aim of narrowing the solution space [45, 65], as illustrated in Fig. 3. These two processes are widely used in the arts and creativity domain to support users in systematically exploring, evaluating, and refining a set of ideas toward a goal. Prior research shows that supporting users in moving between divergent and convergent thinking is key to avoiding fixation [43, 50, 82, 178], a tendency to become anchored to initial ideas, and supports broader, more effective exploration of the solution space [144, 145].

Later on, divergent and convergent thinking have been extended beyond the solution space to structure the broader design process, including problem understanding. One of the most widely adopted instantiations of this extension is the Double Diamond model, popularised by the British Design Council in 2005 [47], as illustrated in Fig. 6. The model structures the design process into four phases across two diamonds: the first diamond addresses the problem space, beginning with *Discover* (diverging to understand the problem broadly rather than assuming what it is) and moving to *Define* (converging to articulate the problem precisely); the second diamond addresses the solution space, beginning with *Develop* (diverging to generate a wide range of candidate solutions) and ending with *Deliver* (converging to test, refine, and release a solution) [47]. Each diamond thus follows the same diverge-then-converge pattern, applied first to the problem and then to the solution. In creative domains such as writing, design, and music composition, this structure is particularly valuable because both the problem and the solution space are multi-dimensional and ill-defined, and cannot be fully anticipated in advance, thus requiring users to explore broadly before they can meaningfully narrow down [56, 92]. For example, a User Experience (UX) designer tasked with improving a mobile banking application may first diverge to interview users, observe their behaviours, and identify a wide range of pain points (Discover), before converging on a precisely defined problem such as users struggling to locate transaction history under time pressure (Define); the designer then diverges again to generate many candidate interface solutions such as a persistent search bar, a timeline view, or a voice query feature (Develop), before converging on a final design through prototyping and user testing (Deliver).

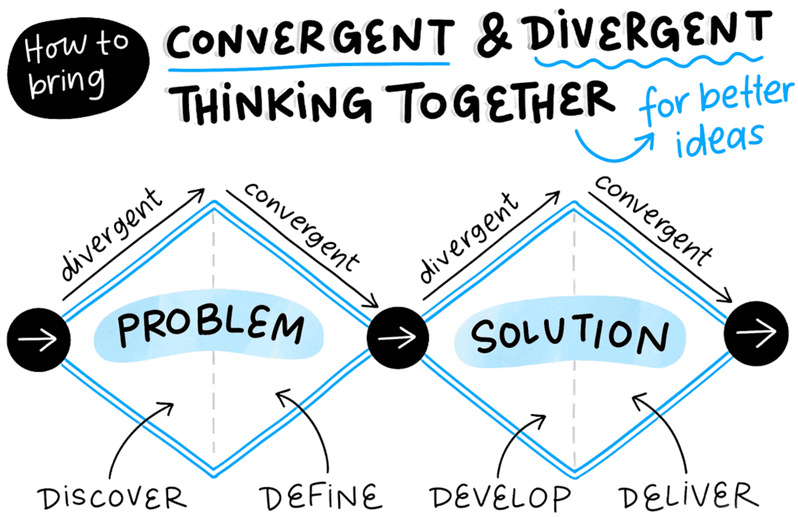


Fig. 6. The Double Diamond model [47], structuring the design process into four phases across two diamonds. The first diamond addresses the problem space: *Discover* (diverging to understand the problem broadly) and *Define* (converging to articulate the problem precisely). The second diamond addresses the solution space: *Develop* (diverging to generate candidate solutions) and *Deliver* (converging to refine and release a solution). Each diamond follows the same diverge-then-converge pattern.

Ontology scoping shares a similar situation to the solution space of the Double Diamond, since ontology engineers approach it with a goal in mind: to define a well-scoped ontology that adequately covers the intended domain [46, 183]. The target ontology scope is, however, multi-dimensional, ill-defined, and cannot be fully anticipated in advance [7, 46, 183, 185]. Ontology engineers must consider at least two key dimensions, subdomains (horizontal) and term granularity (vertical) [46]. Generated CQ candidates populate this space, where each CQ suggests concepts and relations within a specific subdomain and at a specific level of term granularity. Ontology engineers must therefore move between divergent thinking (exploring and generating candidate CQs across subdomains and levels of term granularity) and convergent thinking (refining and deleting CQs along the same dimensions) to progressively define a well-scoped ontology [183]. Reasoning across these dimensions enables ontology engineers to step back from individual CQ candidates and consider the entire candidate space at a higher level [144, 145, 183]: by viewing CQs organised across subdomains and levels of term granularity simultaneously, ontology engineers can identify gaps (subdomains or granularity levels with too few candidates, suggesting where to diverge) and overlaps (subdomains or granularity levels with redundant or conflicting candidates, suggesting where to converge), in order to effectively audit the boundaries of the target ontology, as illustrated in Fig. 7. This connection between ontology scoping and the solution space of divergent and convergent thinking motivates the interaction design focus of this paper.

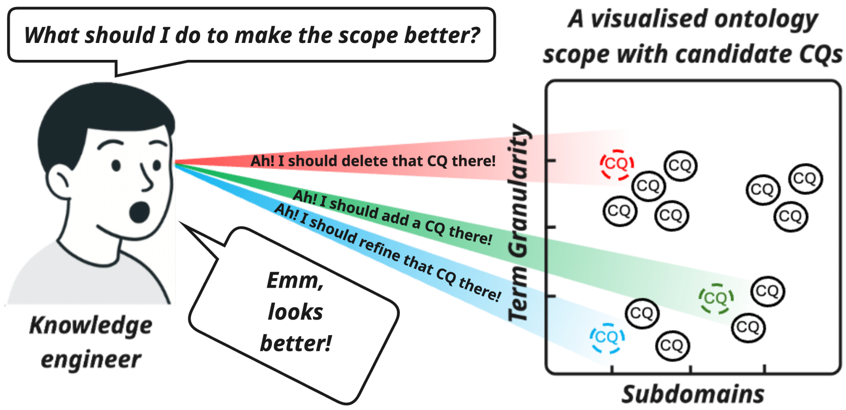


Fig. 7. An ontology engineer reasoning across the dimensions of subdomains (horizontal) and term granularity (vertical) on a visualised ontology scope with candidate CQs, identifying three auditing decisions: **deleting** a redundant or out-of-scope CQ (convergent thinking), **adding** a new CQ to cover a gap (divergent thinking), and **refining** an existing CQ to better fit the intended scope (convergent thinking).

2.4. *Interaction designs supporting divergent and convergent thinking*

To support divergent and convergent thinking, LLM-based systems need to provide users with interaction designs that help them navigate a large, multi-dimensional candidate space: exploring broadly when diverging and narrowing down when converging [144, 145, 183]. In this work, we focus on interaction designs under the screen-mouse-keyboard modality, as this is the most widely adopted modality in LLM-based systems for knowledge work [25]. Under this modality, interaction designs consist of two components. ITs refer to specific user actions that a system supports, such as clicking, dragging, hovering, and scrolling, which determine how users input intentions and how the system responds [81, 177]. UIs refer to the visual and functional components that users interact with, such as buttons, sliders, menus, canvases, and input fields [49, 113]. Together, ITs and UIs determine what a user can do with a system and how the system presents information in response.

A widely adopted framework for organising these interaction designs to support divergent and convergent thinking is Shneiderman's "Visual Information-Seeking Mantra" [136]: *"Overview first, zoom and filter, then details-on-demand"*, as illustrated in Fig. 8. This framework describes a three-stage sequence that guides users from broad exploration toward focused analysis, with each stage capable of supporting both divergent and convergent thinking depending on how users engage with it. In the *Overview* stage [16, 57, 130], the system presents the full candidate space at a high level. This primarily supports *divergent* thinking by enabling users to gain a global understanding of what is available before deciding where to focus, though it can also support *convergent* thinking when users use the overview to identify and prioritise the most relevant regions for further investigation. This stage is supported by ITs such as panning, zooming out, and spatial navigation, and by UIs such as multidimensional spatial canvases, timelines, networks, and hierarchical trees. In the *Zoom and Filter* stage [3, 19, 83, 168], users narrow their focus to a relevant subset of candidates. This supports *convergent* thinking by excluding what is not needed and concentrating attention on what remains, while also supporting *divergent* thinking when users apply exploratory filters to reveal unexpected groupings or patterns within the narrowed space. This stage is supported by ITs such as zooming in, querying, brushing, and faceted selection, and by UIs such as sliders, checkboxes, scroll controls, and search bars. In the *Details-on-Demand* stage [136], users inspect individual candidates in depth without losing sight of the broader context. This primarily supports *convergent* thinking by enabling focused evaluation and verification of specific items, though it can also support *divergent* thinking when detailed inspection surfaces unexpected properties that redirect the user's attention toward new possibilities. This stage is supported by ITs such as hovering, clicking to expand, and drilling down, and by UIs such as tooltips, expandable panels, pop-up windows, and clickable items.

Together, these three stages externalise the structure of the candidate space and provide a flexible scaffold within which both divergent and convergent thinking can be supported across all stages of exploration.

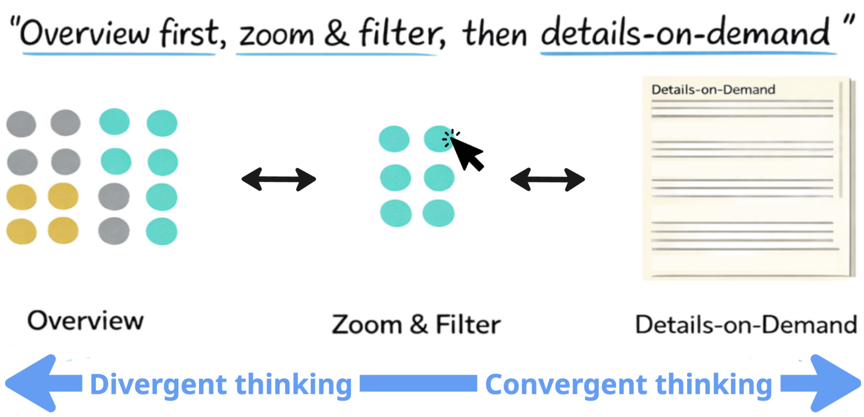


Fig. 8. Shneiderman's "Visual Information-Seeking Mantra" [136]: a three-stage sequence of interaction designs, each capable of supporting both divergent and convergent thinking. *Overview*: the system presents the entire candidate space, primarily supporting broad divergent exploration while also enabling users to identify priorities convergently. *Zoom and Filter*: users narrow focus to a relevant subset, supporting convergent refinement while also enabling divergent discovery through exploratory filtering. *Details-on-Demand*: users inspect individual candidates in depth, supporting convergent evaluation while enabling divergent redirection when unexpected properties are surfaced.

Many existing LLM-based systems [1, 9, 38, 67, 144, 145, 172] follow this framework to support divergent and convergent thinking. However, a systematic understanding of what ITs and UIs are currently used across such systems remains lacking, and how these interaction designs may be applied to support ontology engineers in ontology scoping has yet to be explored.

3. Systematic literature review

We conducted an SLR following the PRISMA 2020 guidelines [118], a widely adopted reporting standard that ensures transparency and reproducibility across the identification, screening, and analysis stages, with additional practical guidance from established SLR methodological works [77, 137, 164]. As this SLR targets a specific intersection of LLM-based systems and interaction design for creative tasks, items in PRISMA 2020 designed for clinical and epidemiological reviews, such as meta-analysis, certainty assessment, and funding bias reporting, are not applicable and are therefore omitted [118, 137]. The full SLR process is illustrated in Fig. 9. Section 3.1 details the identification stage, including search query formulation, database selection, and venue filtering; Section 3.2 outlines the screening stage, including Eligibility Criteria (EC) and IRR assessment; Section 3.3 covers the snowballing stage, including forward and backward reference searches; and Section 3.4 describes the development and validation of the codebook used to analyse the included papers.

3.1. Step 1: identification

We conducted a structured search across three major digital databases: the ACM DL[9], IEEE Xplore[10], and Scopus[11]. These three databases were selected as they collectively provide broad coverage of relevant publication venues in Human–Computer Interaction (HCI), interactive system design, generative AI, and creative computing, including

[9] https://dl.acm.org/
[10] https://ieeexplore.ieee.org/Xplore/home.jsp
[11] https://www.scopus.com/

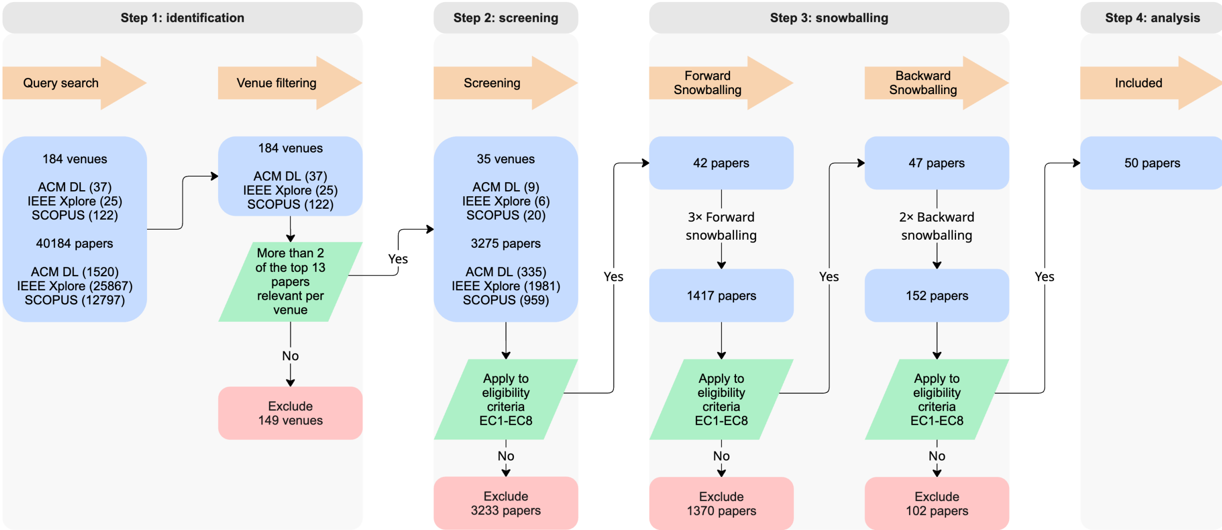


Fig. 9. The SLR pipeline following PRISMA 2020 guidelines [118], consisting of four stages: (1) Identification, in which a query search across three databases (ACM Digital Library (ACM DL), IEEE Xplore, and Scopus) yielded 40,184 papers across 184 venues, reduced to 3,275 papers across 35 venues after venue filtering; (2) Screening, in which EC1–EC8 were applied, retaining 42 papers; (3) Snowballing, consisting of three rounds of forward snowballing and two rounds of backward snowballing, identifying 8 additional papers; and (4) Analysis, resulting in a final dataset of $N = 50$ papers.

the *ACM CHI Conference on Human Factors in Computing Systems* (CHI), the *ACM Symposium on User Interface Software and Technology* (UIST), the *ACM International Conference on Intelligent User Interfaces* (IUI), and the *ACM Designing Interactive Systems Conference* (DIS), all of which are fully indexed by at least one of the three databases. This combination has been consistently adopted in prior SLRs in HCI and interactive AI systems [77, 135], and any papers not retrieved by these databases were expected to be captured through the subsequent snowballing procedure (Section 3.3).

The initial search query was derived from the three core concepts underlying our RQs: (1) LLM-based systems, (2) divergent and convergent thinking, and (3) interaction designs. However, after several trials, we found that restricting to LLM-specific terms retrieved few relevant papers, as LLMs are often integrated as one component within broader multimodal pipelines that delegate tasks such as image generation or audio synthesis to other specialised models [17, 38, 160]; we therefore adopted broader umbrella terms such as `AI` and `machine learning` alongside LLM-specific terms to ensure comprehensive coverage.

We initially included divergent and convergent thinking-specific terms in the search query, as these are the thinking processes that define creative thinking, whose supporting interaction designs this paper seeks to identify. However, we found that including these terms retrieved few additional relevant papers, as these thinking processes are rarely explicitly named in papers that nonetheless implement and evaluate interaction designs supporting them. These terms were therefore removed. We instead retained the focus on arts and creativity as the domain search terms (`art*` and `creati*`), as creative thinking is inherently applied in this domain, and trials confirmed it is where LLM-based interactive systems supporting divergent and convergent thinking are most widely developed and evaluated.

Finally, we found that including interaction-design-related terms such as `interaction techniques` or `user interface` could neither be incorporated via the Boolean operator `OR` nor via the Boolean operator `AND` with the existing two mandatory query components (AI-related terms and arts-related terms). Adding them via `OR` to either component would potentially pair interaction-design-related terms with either AI-related or arts-related terms alone, retrieving a large volume of false positives that would render the scope unmanageable for full-text screening. Adding them via `AND` would require all three components to appear together, substantially reducing recall, as papers describing relevant interaction designs rarely use these terms in titles, abstracts, or keywords (the rationale for restricting the search to these three fields is detailed in the following paragraph) but instead describe them in later sections such as system design, implementation, demonstration, or evaluation. These terms were therefore removed, and the focus on interaction designs was instead maintained through the EC applied at the screening stage (Section 3.2). The final search query was:

```
"artificial intelligence" OR "AI" OR "machine learning" OR
"large language model" OR "large language models" OR LLM*
```

AND

```
generat*
```

AND

```
art* OR creati*
```

We set the publication date range to 2017 onwards, coinciding with the release of the Transformer architecture [154], which marks a key inflexion point in the development of generative models capable of supporting creative tasks [77, 183]. The query was restricted to the *title*, *abstract*, and *keywords* fields, as preliminary trials revealed that searching across full text produced a high rate of false positives, predominantly due to the term `art*` appearing in phrases such as `state-of-the-art` and `state of the art` in unrelated papers. Following these settings, the search yielded 40,184 non-duplicate records across 184 venues from the three databases.

To make the resulting dataset of 40,184 papers manageable while preserving coverage of relevant work, we applied venue filtering as a precursor to manual full-text screening [77, 137, 164]. We assessed venue relevance by examining the top 13 retrieved papers per venue based on full text review against the EC, detailed in Section 3.2, retaining a venue if more than 2 of those 13 papers were deemed relevant. The threshold of 13 was selected as it corresponds to the smallest number of papers retrieved from any single venue in our dataset, ensuring a consistent basis for comparison across all venues. The cut-off of more than 2 was determined through a sensitivity analysis: compared to a cut-off of 1, it improved dataset manageability; compared to a cut-off of 3, it retained more venues, better preserving coverage of relevant papers. We acknowledge that venue filtering may introduce limitations by excluding papers published in venues with lower overall relevance to our topic; however, this limitation is mitigated by the subsequent snowballing stage (Section 3.3), which is designed to recover relevant papers not captured during the initial search. Following venue filtering, 149 venues were excluded, and 3,275 papers across 35 venues were retained for the screening stage.

3.2. *Step 2: screening*

To screen the 3,275 papers, we defined eight EC organised into two groups, as detailed in Table 1. EC-1 to EC-4 are general quality criteria ensuring that only peer-reviewed, non-retracted, non-secondary, English-language primary studies were retained. EC-5 to EC-8 are topic-specific criteria that assess whether a paper is relevant to our research focus. We did not restrict paper types to full papers during screening, as short papers, workshop papers, demos, and symposium papers can also present relevant interaction designs, though those without evaluation of their effectiveness in supporting the creative process would be subsequently excluded by EC-8. The eight criteria were applied sequentially, with EC-1 to EC-4 applied first to eliminate papers on general quality grounds before applying the more demanding topic-specific criteria EC-5 to EC-8.

The first two authors independently screened a random sample of 20% ($N = 655$) of the papers in full text to calibrate their application of the EC. IRR was calculated using Cohen's kappa coefficient prior to any coordinating discussions, yielding $\kappa = 0.82$, indicating "Strong" agreement (0.80–0.90) [104]. This level of agreement is attributable to two factors: both authors have backgrounds in HCI and interactive system design with more than three years of experience; and the criteria are primarily concerned with identifying whether a paper presents an LLM-based system with clearly defined interaction designs that have been evaluated, a judgement directly assessable from the system design, implementation, and evaluation sections of each paper.

Disagreements were resolved through discussion until consensus was reached, primarily concerning `EC-5`: determining whether the outputs generated by an LLM-based system constitute an instance of artistic expression or

Group	EC-#	Description
General quality	`EC-1`	The full text of the paper must be available in English.
	`EC-2`	The paper must not have been retracted.
	`EC-3`	The paper must be peer-reviewed.
	`EC-4`	The paper must not be a secondary study (e.g., a survey).
Topic-specific	`EC-5`	The paper must present an LLM-based system that supports creative processes in artistic expression or creative design.
	`EC-6`	The LLM-based system must incorporate ITs that support the creative process.
	`EC-7`	The LLM-based system must incorporate UIs that support the creative process.
	`EC-8`	The interaction designs incorporated in the paper must be evaluated to show their effectiveness in supporting the creative process.

Table 1

EC used in the screening process, organised into general quality criteria (EC-1 to EC-4) and topic-specific criteria (EC-5 to EC-8).

creative design. The main source of ambiguity was distinguishing between Procedural Content Generation (PCG) systems, where the user sets parameters such as difficulty level or generation seed and the system autonomously produces the output according to predefined rules without any further user involvement in shaping the result, and creative support systems, where the user actively shapes the expressive or aesthetic qualities of the output through iterative interaction, such as refining the style, tone, or composition of a generated artefact. The key distinction is therefore whether the system supports users in iteratively shaping the expressive or aesthetic qualities of the output, rather than merely configuring parameters that determine how content is algorithmically produced. Systems that did not meet `EC-5` on this basis were also excluded at `EC-6` and `EC-7`, as their ITs and UIs are intended to configure algorithmic parameters rather than to support expressive user decision-making over the creative outcome.

The remaining 80% of papers were divided equally between the two authors (each screening $N = 1,310$ papers independently), with uncertain cases flagged and resolved at a final consensus meeting. A large number of papers were excluded at `EC-6`, as they focused on fully automated generation pipelines or model-centric optimisation without clearly defined interaction designs. After applying all criteria, $N = 42$ papers were retained for the snowballing stage.

3.3. Step 3: snowballing

To ensure comprehensive coverage of relevant work not captured during the initial database search, we conducted snowballing. Three iterations of forward snowballing were conducted using Google Scholar[12] to retrieve papers citing each included paper, with an initial title and abstract screen applied to filter out papers not mentioning AI-based system design for arts and creativity, yielding 1,417 candidate papers. The first two authors then independently screened the full text of remaining candidates against the EC-1 to EC-8 (Section 3.2), each screening half of the papers, with uncertain cases flagged and resolved through discussion at a final consensus meeting, resulting in $N = 5$ additional papers being identified. Two iterations of backward snowballing were subsequently conducted by examining the reference lists of all included papers with the same initial screen, yielding 152 candidate papers, which were screened using the same process, resulting in $N = 3$ additional papers being identified. Combined with the $N = 42$ papers retained from the screening stage, forward and backwards snowballing contributed an additional $N = 8$ papers, bringing the final corpus to $N = 50$ papers, with publication years ranging from 2021 to 2025.

We prioritised forward snowballing as our research focus targets a rapidly evolving field in which the most relevant recent work predominantly cites other recent contributions, meaning that forward snowballing from already-included papers is more likely to surface additional relevant work than starting from older references, and the newly

[12]https://scholar.google.com/

identified papers from forward snowballing in turn enriched the subsequent backward snowballing with a broader set of reference lists to trace. We acknowledge that this deviates from the conventional backward-forward sequence [171], which may have resulted in fewer relevant foundational papers being identified and available for inclusion in the forward search; however, we conducted three iterations of forward snowballing followed by two iterations of backward snowballing until no additional papers could be identified in either process, minimising the impact of this deviation on overall coverage.

3.4. Step 4: analysis

To analyse the 50 papers and address our research questions, we developed a codebook using a hybrid approach combining top-down and bottom-up strategies [29, 54], as this allows theoretical grounding from established frameworks to be balanced with emergent insights from the data. The first two authors developed the codebook together synchronously in person, beginning with a top-down strategy guided by established taxonomies of UIs [28] and ITs [30] to define the initial structure and first-level categories. They then applied a bottom-up approach by conducting open coding on a randomly selected subset of 50% ($N = 25$) of the papers to refine the first-level categories and inductively develop second-level categories, which served as the final codes for the analysis. The codebook was developed synchronously in person, as this allowed discrepancies to be discussed immediately and a shared understanding of the codebook structure and coding decisions to be established before independent coding began.

After finalising the codebook, the first two authors independently applied it to the remaining 50% ($N = 25$) of the papers, assigning binary values to denote the presence (1) or absence (0) of each code for each paper. IRR was calculated using Cohen's kappa coefficient for each individual code prior to any coordinating discussions, yielding an average score of $\kappa = 0.85$, indicating "Strong" agreement (0.80–0.90) [104]. This level of agreement is attributable to the synchronous in-person codebook development process, which established a shared and precise understanding of each code prior to independent coding, and to the binary nature of the coding scheme, which reduces ambiguity in coding decisions compared to ordinal or multi-label schemes. Disagreements were resolved through discussion until consensus was reached, primarily concerning whether the use of a slider should be considered a simple parameter adjustment or an interpolative behaviour across multiple variables [10, 89, 153].

3.5. Results

This section presents the analysis results of the 50 papers retained following the search, screening, and snowballing stages. We describe the distribution of papers across publication years in Section 3.5.1, venues in Section 3.5.2, creative domains in Section 3.5.3, ITs in Section 3.5.4, and UIs in Section 3.5.5.

3.5.1. Publication trends

Table 2 presents the distribution of the 50 retained papers by publication year. The corpus spans 2021–2025, with a notable concentration in 2023 ($N = 19$, 38%) and 2024 ($N = 21$, 42%), together accounting for 80% of all retained papers, which may be attributable to the widespread availability of large-scale generative models following the public release of ChatGPT[13] in late 2022, which substantially lowered the barrier to integrating LLM capabilities into interactive system prototypes evaluated in user studies. The three papers from 2025 reflect the fact that the search was conducted in early 2025 and therefore captured only a partial year of publications.

3.5.2. Publication venues

Table 3 presents the distribution of the 50 retained papers across publication venues. The corpus is predominantly drawn from HCI and interactive systems venues, with the CHI accounting for the largest share ($N = 19$, 38%), followed by the UIST ($N = 8$, 16%), the IUI ($N = 6$, 12%), and the DIS ($N = 3$, 6%). Together, these four venues account for 72% of the retained corpus, and HCI and interactive systems venues as a whole account for 80% ($N = 40$).

The predominance of CHI, UIST, IUI, and DIS papers in the retained corpus provides validation of the database selection described in Section 3.1. The three databases were selected on the assumption that these four venues are

[13] https://en.wikipedia.org/wiki/ChatGPT

Table 2
Distribution of retained papers by publication year.

Year	*N*	**%**
2021	2	4%
2022	5	10%
2023	19	38%
2024	21	42%
2025	3	6%
Total	**50**	**100%**

Table 3
Distribution of retained papers by publication venue, grouped into three research communities.

Community	**Venue**	*N*	**%**
HCI and interactive systems	ACM CHI Conference on Human Factors in Computing Systems (CHI)	19	38%
	ACM Symposium on User Interface Software and Technology (UIST)	8	16%
	ACM International Conference on Intelligent User Interfaces (IUI)	6	12%
	ACM Designing Interactive Systems Conference (DIS)	3	6%
	ACM Transactions on Computer-Human Interaction	1	2%
	Proceedings of the ACM on Human-Computer Interaction	1	2%
	ACM Collective Intelligence Conference Series	1	2%
	International Academic Mindtrek Conference	1	2%
AI and natural language processing	AAAI Artificial Intelligence and Interactive Digital Entertainment	2	4%
	Annual Meeting of the Association for Computational Linguistics	1	2%
	Conference of the European Chapter of the ACL	1	2%
	Conference on Empirical Methods in Natural Language Processing	1	2%
	Conference on Neural Information Processing Systems	1	2%
Domain-specific venues	International Society for Music Information Retrieval Conference	1	2%
	International Conference on the Foundations of Digital Games	1	2%
	IEEE Visualization and Visual Analytics	1	2%
	International Conference on Data Science and Management of Data	1	2%
Total		**50**	**100%**

representative publication outlets for the survey topic and are fully indexed by the selected databases; however, whether these venues would in practice account for the majority of relevant papers could only be determined at this stage. The finding that 72% of retained papers originate from these four venues confirms both that these venues are representative of the field and that the database selection was appropriate for the purpose of this survey.

The remaining 10 papers (20%) are distributed across AI and natural language processing venues ($N = 6$, 12%) and domain-specific venues ($N = 4$, 8%) spanning music information retrieval, game studies, information visualisation, and data science.

3.5.3. *Creative domains*

Table 4 presents the distribution of the 50 retained papers across six creative domains. As noted in Section 3.1, the retained papers predominantly feature systems in which an LLM is integrated as a central component within broader multimodal pipelines for tasks such as prompt generation, semantic guidance, and narrative structuring, with domain-specific outputs delegated to specialised models (e.g., Stable Diffusion, DALL-E).

Creative writing is the most represented domain ($N = 19$, 38%), referring to the composition of textual narratives, stories, and expressive written content, which may reflect the relative maturity of LLM capabilities in text generation. Visual design ($N = 14$, 28%) refers to the creation of static visual outputs through the deliberate use of composition, colour, typography, and imagery to communicate meaning or aesthetic intent. Video creation ($N = 8$, 16%) refers to the production of temporally sequenced visual and narrative content. Game design ($N = 4$, 8%) refers to the authoring of game content such as narratives, characters, and level design through iterative expressive choices, rather than PCG systems where content is algorithmically produced according to predefined rules or parameters without meaningful user agency over the expressive or aesthetic qualities of the result. 3D and spatial design ($N = 3$, 6%) refers to the creation and manipulation of three-dimensional objects, environments, and spatial configurations. Music creation, referring to the composition and production of audio content, and creative coding, referring to the use of programming as a medium for artistic expression, each accounts for a single paper ($N = 1$, 2%), suggesting potential directions for future work in their respective communities.

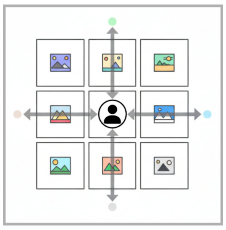
Table 4

Distribution of retained papers by creative domain.

Creative domain	*N*	%
Creative writing	19	38%
Visual design	14	28%
Video creation	8	16%
Game design	4	8%
3D and spatial design	3	6%
Music creation	1	2%
Creative coding	1	2%
Total	**50**	**100%**

3.5.4. *Interaction techniques*

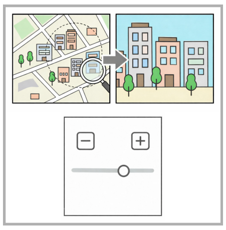
For each IT, we describe what it is with reference to representative systems from our corpus, justify how it structurally supports divergent thinking, convergent thinking, or both by examining whether it enables users to broaden the range of ideas considered towards a goal, narrow attention to refine and evaluate options towards the same goal, or facilitates transitions between these two modes [66], and discuss how LLM-based systems, combined with multimodal models, can be used to augment the thinking processes it supports. Additionally, IRR for each identified category was assessed using Cohen's κ [104], with scores reported and colour-coded per entry: **above 0.90 (Almost Perfect)**, **0.80–0.90 (Strong)**, and **0.60–0.79 (Moderate)**.



Spatial Navigation

Cohen's κ = **0.84** (Strong)

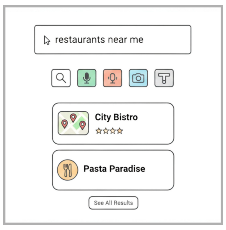
IT-1 Spatial navigation allows users to traverse a multi-dimensional design space through directional controls, as seen in Spellburst [10] and Luminate [145]. Because users explore by moving through the space rather than formulating a query, they are exposed to content they would not have thought to ask for, which broadens the range of ideas considered and supports *divergent* thinking. When a promising region is found, navigating within it allows users to progressively narrow down options, supporting *convergent* thinking. LLM-based systems, combined with multimodal models, can be used to populate each spatial region with contextually relevant text, image, or mixed-media content in response to the user's current position, ensuring that every navigational step surfaces meaningful and varied possibilities [27, 97, 157, 179].



Zooming

Cohen's κ = **0.92** (Almost Perfect)

IT-2 Zooming allows users to adjust the scale of visible information between global overview and fine-grained detail, as seen in PromptMap [1] and Spellburst [10]. Because users can pull back to see the full landscape of possibilities without committing to any, zooming out supports *divergent* thinking by revealing the breadth of the space and encouraging broad exploration. Zooming in then supports *convergent* thinking by focusing attention on a specific region for detailed inspection and refinement. LLM-based systems, combined with multimodal models, can be used to generate high-level textual summaries and visual overviews when zoomed out, and fine-grained textual elaborations alongside detailed image content when zoomed in, adapting the richness of output to the user's current level of focus [9, 27, 34, 37, 181].



Querying

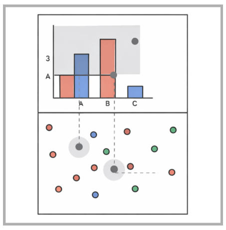
Cohen's κ = **0.76** (Moderate)

IT-3 Querying allows users to specify information needs through prompts or keywords, as seen in Wordcraft [179] and PromptMap [1]. Because users can issue open-ended queries without a precise target, querying supports *divergent* thinking by surfacing semantically related content that broadens the user's conception of what is possible. As queries become more specific and targeted, the technique supports *convergent* thinking by progressively narrowing the field and sharpening conceptual direction toward a defined goal. LLM-based systems, combined with multimodal models, can be used to interpret the same query across text and visual modalities simultaneously, returning semantically relevant textual suggestions alongside generated image or mixed-media content that together give users a richer and more complete view of the possibility space [10, 27, 34, 107, 186].

Sorting and Filtering

Cohen's κ = **0.84** (Strong)

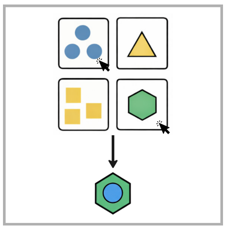
IT-4 Sorting and filtering allow users to reorganise or constrain visible content based on defined criteria, as seen in ExpressEdit [153] and Metaphorian [88]. Because filtering removes irrelevant content while sorting surfaces patterns across the remaining items, the technique supports *convergent* thinking by helping users prioritise salient information and focus attention on what matters most. When users apply loose or exploratory criteria to discover unexpected groupings, it can also support *divergent* thinking by revealing structural patterns in the possibility space that were not previously apparent. LLM-based systems, combined with multimodal models, can be used to propose semantically and visually meaningful sorting and filtering strategies across both text and image content, helping users impose structure on heterogeneous generated outputs without needing to define criteria manually [88, 98, 139, 150, 159, 179].



Linking and Brushing

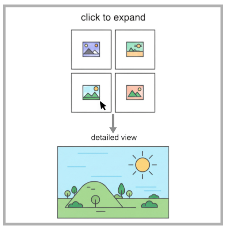
Cohen's κ = **0.76** (Moderate)

IT-5 Linking and brushing enable users to select a subset of elements and immediately highlight semantically associated items across multiple views, as seen in Luminate [145] and Datatales [146]. Because selecting one element reveals unexpected connections to others, it supports *divergent* thinking by surfacing thematic relationships that users would not have identified through sequential inspection alone. As users refine their selections to isolate a coherent cluster of related items, it supports *convergent* thinking by helping build integrated cognitive structures across representations. LLM-based systems, combined with multimodal models, can be used to infer latent semantic and visual connections across both text and image content simultaneously, ensuring that brushing a textual element surfaces not only related text but also visually associated materials, giving users a unified and cross-modal view of how ideas relate [1, 27, 37, 98, 107, 146].



Clustering and Combining
Cohen's κ = **0.88** (Strong)

IT-6 Clustering and combining allow users to group or merge related elements into higher-order constructs, as seen in CreativeConnect [37] and AutoSpark [34]. Because grouping distributed ideas into clusters requires surveying the full possibility space first, this technique supports *divergent* thinking by externalising emerging patterns that were not initially apparent. When clusters are merged and consolidated into coherent conceptual units, the technique supports *convergent* thinking by reducing complexity and synthesising scattered information toward a unified structure. LLM-based systems, combined with multimodal models, can be used to identify semantic relatedness across text and visual content simultaneously, proposing groupings and consolidation strategies that span both modalities and make latent conceptual structure visible across heterogeneous materials [1, 9, 27, 78, 145].

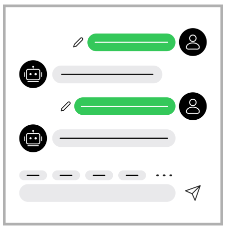


Mouse Hover and Click-to-Expand
Cohen's κ = **0.72** (Moderate)

IT-7 Mouse hover and click-to-expand provide users with lightweight preview access or persistent detailed views of generated content, as seen in Spellburst [10] and Patchview [38]. Because hovering allows users to quickly scan across many items without committing to any, it supports *divergent* thinking by lowering the cost of exploration and enabling rapid comparison across a wide range of possibilities. Clicking to expand then supports *convergent* thinking by giving users persistent access to detailed content for critical evaluation and interpretation verification toward a decision. LLM-based systems, combined with multimodal models, can be used to generate concise textual summaries alongside visual thumbnails for hover previews, and richer textual elaborations with full image or mixed-media content for expanded views, giving users a coherent and layered understanding of each item at both levels of detail [27, 97, 98, 157, 158, 187].

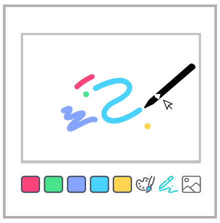
3.5.5. User interface designs

For each UI, we describe what it is with reference to representative systems from our corpus, justify how it structurally supports divergent thinking, convergent thinking, or both by examining whether it enables users to broaden the range of ideas considered towards a goal, narrow attention to refine and evaluate options towards the same goal, or facilitates transitions between these two modes [66], and discuss how LLM-based systems, combined with multimodal models, can be used to augment the thinking processes it supports. Additionally, IRR for each identified category was assessed using Cohen's κ [104], with scores reported and colour-coded per entry: **above 0.90 (Almost Perfect)**, **0.80–0.90 (Strong)**, and **0.60–0.79 (Moderate)**.



Chatbot
Cohen's κ = **0.92** (Almost Perfect)

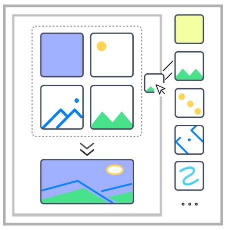
UI-1 Chatbot structures interaction into manageable conversational turns, as seen in Wordcraft [179] and CALYPSO [187]. Because each turn surfaces a new response without requiring users to plan ahead, it supports *divergent* thinking by allowing ideas to emerge incrementally through dialogue rather than through deliberate query design. As the conversation progresses and responses become more targeted, it supports *convergent* thinking by maintaining attention on immediate reasoning goals and segmenting complex thinking into focused steps. LLM-based systems, combined with multimodal models, can be used to generate contextually relevant replies and proactively suggest next prompts while also producing image or mixed-media responses within the same conversational structure, helping users sustain cognitive flow across turns regardless of modality [100, 107, 145, 158, 175, 186].



Canvas

Cohen's κ = **0.84** (Strong)

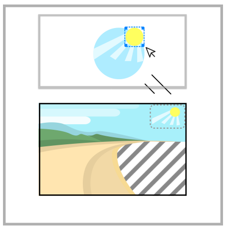
UI-2 Canvas provides a flexible spatial workspace where users can freely place, arrange, and connect generated content, as seen in Spellburst [10] and TaleBrush [39]. Because the open-ended nature of the space imposes no predefined structure, it supports *divergent* thinking by allowing users to externalise fragmented ideas freely and discover relationships through spatial arrangement rather than sequential reasoning. As users progressively organise and stabilise content on the canvas, it supports *convergent* thinking by making emerging mental representations visible and refinable. LLM-based systems, combined with multimodal models, can be used to propose spatial annotations, suggest connections between placed elements, and respond to sketch or drawing inputs to generate image content directly within the canvas, creating a unified workspace where textual and visual generation reinforce each other [10, 39, 145, 174].



Moodboard

Cohen's κ = **0.88** (Strong)

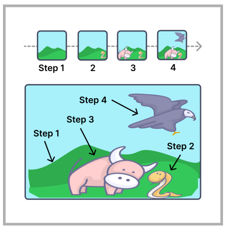
UI-3 Moodboard enables users to openly arrange heterogeneous materials in a shared visual space, as seen in CreativeConnect [37] and Lotus [17]. Because users can place diverse and loosely related items without committing to a structure, it supports *divergent* thinking by encouraging associative reasoning and thematic exploration across materials that would not naturally be considered together. As implicit patterns emerge from the arrangement, it supports *convergent* thinking by helping users abstract recurring themes and organise fragmented inputs into coherent groupings. LLM-based systems, combined with multimodal models, can be used to identify clustering patterns across both textual and visual items, suggest semantic labels that make latent structure explicit, and generate image content that fills thematic gaps in the emerging groupings [1, 27, 34, 78, 103, 162].



Focus Region in Context

Cohen's κ = **0.76** (Moderate)

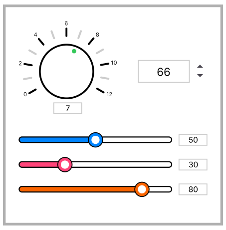
UI-4 Focus region in context links a locally selected area to its surrounding global view, allowing users to edit or inspect detail while retaining awareness of the broader structure, as seen in Directgpt [103] and Luminate [145]. Because the global view remains visible during local operations, it supports *divergent* thinking by allowing users to identify new areas of interest across the full space while working on a specific region. When users apply focused edits that propagate meaningfully to the global structure, it supports *convergent* thinking by helping users maintain cognitive coherence across hierarchical levels and progressively refine their overall output. LLM-based systems, combined with multimodal models, can be used to generate context-sensitive textual refinements for the focused region while simultaneously applying localised image edits that remain visually consistent with the surrounding global content [76, 78, 153, 162, 174].



Spatially Structured and Sequentially Guiding Editor

Cohen's κ = **0.84** (Strong)

UI-5 Spatially structured and sequentially guiding editors externalise task organisation through grids, timelines, or ordered slots that impose a scaffold on the working space, as seen in Coauthor [93] and DreamDirector [186]. Because the scaffold makes the overall structure visible from the start, it supports *divergent* thinking by freeing users to populate individual slots or cells with varied content without losing sight of how each contribution fits into the whole. As users fill in the structure incrementally, it supports *convergent* thinking by reducing planning overhead and guiding users toward a temporally or spatially coherent final representation. LLM-based systems, combined with multimodal models, can be used to propose context-aligned textual completions for each slot while generating image or video content that fits within the visual and temporal constraints of the grid or timeline, ensuring coherence across both text and image throughout the emerging structure [98, 108, 153, 157, 181, 186].

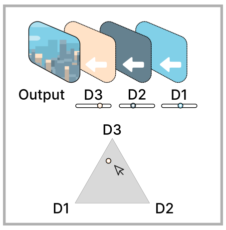


Slider, Knob, and Numeric Textfield
Cohen's κ = **0.96** (Almost Perfect)

UI-6 Sliders, knobs, and numeric textfields allow users to vary parameter values with fine-grained precision, as seen in Promptcharm [162] and Spellburst [10]. Because continuous adjustment allows users to sweep across a parameter range and observe how outputs change, it supports *divergent* thinking by enabling systematic exploration of the effect space without requiring explicit reformulation of intent. When users identify a meaningful range and converge on a specific value, it supports *convergent* thinking by giving precise control over subtle adjustments that shape the final output. LLM-based systems, combined with multimodal models, can be used to generate real-time semantic previews in both text and visual form as parameter values change, giving users a coherent and immediately perceivable understanding of how adjustments affect the output across modalities [17, 27, 89, 116, 159, 179].

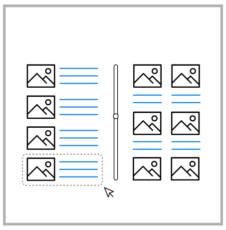
Binary and Likert-Scale Rating
Cohen's κ = **0.92** (Almost Perfect)

UI-7 Binary and Likert-scale rating compress subjective judgements into fixed scales, as seen in Help Me Write a Poem [32]. Because rating options are presented without requiring users to articulate their reasoning, it supports *convergent* thinking by helping users quickly consolidate evolving impressions into explicit evaluative decisions and reduce cognitive ambiguity during assessment. LLM-based systems, combined with multimodal models, can be used to generate micro-explanations alongside rating options that clarify the distinction between adjacent scale points across both textual and visual content, reducing uncertainty and helping users commit to a judgement with greater confidence [32].



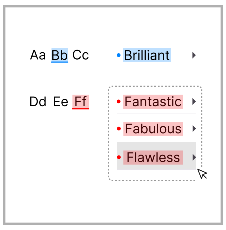
Interpolating Slider, Region, and Graph
Cohen's κ = **0.76** (Moderate)

UI-8 Interpolating controls allow users to blend properties along continuous dimensions, producing outputs that gradually transition between two or more defined states, as seen in Cells, Generators, and Lenses [89] and Patchview [38]. Because users can sweep across the full interpolation range to observe how outputs evolve between extremes, it supports *divergent* thinking by enabling exploration of the variant space between known reference points, surfacing intermediate possibilities that would not have been considered independently. When users identify a preferred region along the continuum and settle on a specific blend, it supports *convergent* thinking by reinforcing comparative reasoning and guiding gradual preference formation. LLM-based systems, combined with multimodal models, can be used to generate semantic textual snapshots at intermediate points along the interpolation path while simultaneously rendering visually interpolated image outputs, making the transition between variants interpretable and directly perceivable across both text and image [38, 78, 89].



List and Grid
Cohen's κ = **0.84** (Strong)

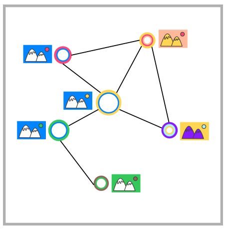
UI-9 List and grid layouts structure the presentation of generated items in a systematic and scannable format, as seen in Luminate [145] and AutoSpark [34]. Because all items are visible simultaneously without requiring navigation, it supports *divergent* thinking by enabling users to survey a broad range of possibilities at once and make connections across items that would not emerge from sequential presentation. When users scan to identify the most relevant candidates and narrow their selection, it supports *convergent* thinking by focusing selective attention and reducing information overload. LLM-based systems, combined with multimodal models, can be used to highlight contextually pertinent candidates and populate grids with both textual and image or mixed-media outputs, enabling users to direct attention toward the most promising items while making visual and semantic comparison across the full set immediately [1, 27, 34, 97, 98, 107].



In-flow Options

Cohen's κ = **0.84** (Strong)

UI-10 In-flow options introduce context-sensitive suggestions directly within the user's ongoing interaction, without requiring them to pause or switch context, as seen in Wordcraft [179] and Spellburst [10]. Because suggestions appear at the point of action and reflect the current context, it supports *divergent* thinking by surfacing branching possibilities that users would not have considered independently, stimulating associative thinking without interrupting ideation flow. When users select a suggestion that aligns with their emerging intent, it supports *convergent* thinking by accelerating decision-making at moments of uncertainty. LLM-based systems, combined with multimodal models, can be used to predict likely next steps and generate contextually relevant branching options in both text and visual form, inline within the same interaction flow, keeping the range of possibilities visible without overwhelming the user [27, 100, 162, 179, 181, 186].



Node-based Branches

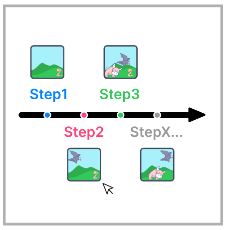
Cohen's κ = **0.92** (Almost Perfect)

UI-11 Node-based branch interfaces externalise conceptual networks spatially, allowing users to build and navigate non-linear idea structures through connected nodes, as seen in Spellburst [10] and PromptMap [1]. Because each node can branch in multiple directions without disrupting existing paths, it supports *divergent* thinking by allowing users to explore several idea trajectories simultaneously and make non-linear associations visible. As users prune, reorganise, and consolidate branches into a coherent hierarchy, it supports *convergent* thinking by helping users systematically refine evolving idea structures toward a resolved output. LLM-based systems, combined with multimodal models, can be used to suggest semantically relevant expansions at each node and generate both textual and visual content that enriches the network, ensuring that the branching structure reflects meaningful conceptual relationships rather than arbitrary connections [1, 10, 34, 38, 78, 181].

Map and Earth

Cohen's κ = **0.96** (Almost Perfect)

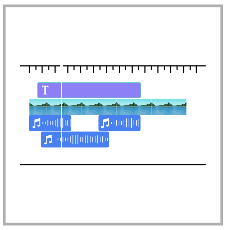
UI-12 Map and earth interfaces enable users to explore content spatially across multiple scales, as seen in Promptify [27] and A Redhead Walks into a Bar [58]. Because the full spatial extent of the content is visible at once, it supports *divergent* thinking by allowing users to identify and navigate toward distant or overlooked conceptual territories without losing awareness of the broader spatial organisation. As users zoom into a specific region and begin working within it, it supports *convergent* thinking by focusing attention on a bounded area for detailed exploration and refinement. LLM-based systems, combined with multimodal models, can be used to highlight semantically promising regions across the map and generate region-specific textual and visual content as users navigate, ensuring that relevant materials across both modalities are surfaced in proportion to the user's current spatial focus [27, 58].



Timeline, Storyline, and Step Sequence

Cohen's κ = **0.84** (Strong)

UI-13 Timeline, storyline, and step sequence interfaces present content within an ordered temporal or procedural structure, as seen in Visar [181] and DreamDirector [186]. Because the structure makes the full sequence visible at once, it supports *divergent* thinking by allowing users to identify gaps, alternative orderings, or unexplored branches across the entire arc without losing track of the overall progression. As users refine individual steps and strengthen transitions between them, it supports *convergent* thinking by incrementally improving narrative logic and temporal cohesion toward a complete and coherent output. LLM-based systems, combined with multimodal models, can be used to suggest chronologically appropriate textual completions for empty steps and generate visual content that fills gaps within the temporal structure, while also proposing missing transitions that maintain consistency across both text and image content throughout the sequence [17, 39, 153, 157, 181, 186].



Lanes and Tracks

Cohen's κ = **0.88** (Strong)

UI-14 Lanes and tracks layer multiple content streams across a shared time axis, allowing users to view and edit concurrent developments in parallel, as seen in LAVE [158] and PodReels [160]. Because all streams are visible simultaneously, it supports *divergent* thinking by enabling users to identify cross-stream opportunities and explore how independent threads might interact or diverge across time. As users align and coordinate content across lanes to ensure consistency, it supports *convergent* thinking by helping maintain cognitive coherence across interdependent sequence layers. LLM-based systems, combined with multimodal models, can be used to propose cross-stream alignments, flag temporal inconsistencies between lanes, and synchronise textual and visual or audio streams within the same track structure, reducing the cognitive effort of managing multiple concurrent threads across modalities [17, 158, 160].

4. Design thinking workshop

This workshop was conducted in person to translate the ITs and UIs identified in Section 3.5 into an interaction model for LLM-based ontology scoping with CQs, implement it as a working system prototype, and explore its applicability through a practical use case. The design thinking process [68], as illustrated in Fig. 10, was selected for this purpose because (1) it begins with an Empathise stage that ensures this translation is grounded in the actual challenges faced by ontology engineers in LLM-based ontology scoping rather than assumed ones; (2) it brings together OE experts and HCI researchers to ensure that design decisions are jointly informed by the knowledge necessary to produce an interaction model capable of addressing those challenges; and (3) its stage-by-stage structure grounds each design decision in the preceding stage, making the full chain of reasoning from challenge identification to evaluation transparent and traceable.

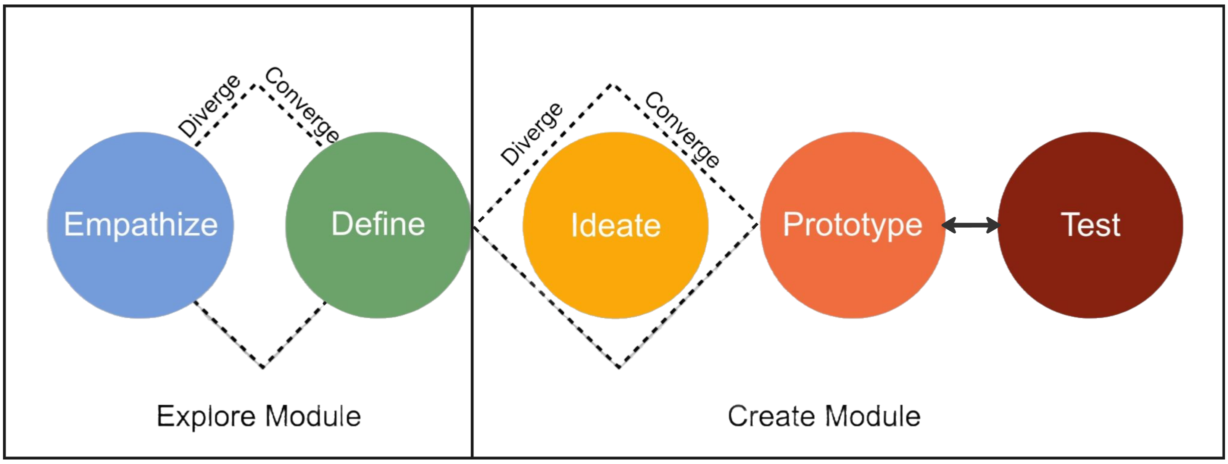


Fig. 10. The design thinking process [68] adopted in this workshop is structured across five stages organised into two modules. The Explore Module comprises Empathise, which supports divergent thinking by identifying the actual challenges faced by ontology engineers in LLM-based ontology scoping with CQs, and Define, which supports convergent thinking by synthesising those challenges into focused problem statements. The Create Module comprises Ideate, which encompasses both divergent thinking in exploring how each problem statement could be addressed through the application of specific ITs and UIs, and convergent thinking in selecting and integrating those that best address the problem statements into a single coherent interaction model, followed by Prototype, which produces a visual representation of the interaction model and implements it as a working system prototype, and Test, which explores the prototype's applicability through a practical use case demonstration. Prototype and Test are conducted iteratively, with one complete cycle performed in this workshop.

This workshop produced three outputs: a conceptual interaction model for LLM-based ontology scoping with CQs (Table 7, Fig. 11), a system prototype implementing that model called OntoScope (Section 4.3, Fig. 12, Fig. 13), and a use case demonstration exploring OntoScope's applicability in practice (Section 4.5). Participants are introduced in Section 4.1, and the workshop procedure is described across two sessions: Session 1 covering Empathise, Define, Ideate, and Prototype (Section 4.2), and Session 2 covering Test (Section 4.4).

4.1. Participants

This workshop involved 7 participants in total. The first and second authors (A1, A2) also participated in all workshop stages as domain experts, each with expertise in both OE and HCI; in addition to their participation, A1 served as facilitator and A2 served as note-taker. We acknowledge the potential facilitator and participant bias introduced by A1 and A2; however, this is acceptable for three reasons: first, A1 and A2 conducted the SLR reported in Section 3.5 and are therefore the most qualified to explain the identified ITs and UIs and contextualise their relevance to LLM-based ontology scoping with CQs, making their active participation a necessity; second, A1's facilitator role was procedural, managing discussion structure and time rather than directing design decisions; and third, all design decisions were reached through structured discussion to consensus among all 7 participants, ensuring the outputs are attributable to collective deliberation rather than to the authors alone.

The remaining 5 participants (P1–P5) were recruited through purposive sampling from 2 research groups at King's College London: the Human-Centred Computing Research Group and the Knowledge Graph Research Group. Purposive sampling was selected because the workshop required participants with verified expertise in either ontology scoping with CQs or interaction design supporting divergent and convergent thinking, and such expertise is concentrated within a small number of specialist research groups, making targeted recruitment more appropriate than open calls. Ethics approval was obtained from King's College London, under which no personal identifying information was collected; all recruited participants were assigned anonymous participant identifiers (P1–P5). All participants volunteered and provided informed consent prior to participation.

Table 5

Demographic and self-reported familiarity data for the 5 recruited participants (P1–P5). OE familiarity questions were answered by OE experts (P1, P2, P3); HCI familiarity questions were answered by HCI researchers (P4, P5). Familiarity was self-reported on a 5-point Likert scale (1 = Not familiar at all, 2 = Slightly familiar, 3 = Moderately familiar, 4 = Familiar, 5 = Very familiar). – indicates the question was not applicable to the participant.

	P1	P2	P3	P4	P5
Role	OE expert	OE expert	OE expert	HCI researcher	HCI researcher
Experience years in OE	8	5	5	–	–
Experience years in HCI	–	–	–	9	7
Gender	Female	Male	Male	Female	Male
Academic level	Postdoc	PhD Candidate	PhD Candidate	Postdoc	PhD Candidate
OE Familiarity					
Familiarity with requirements elicitation in OE	5	5	5	–	–
Familiarity with use of CQs for requirements elicitation in OE	5	4	4	–	–
Familiarity with use of LLMs for requirements elicitation with CQs in OE	4	3	4	–	–
HCI Familiarity					
Familiarity with divergent and convergent thinking	–	–	–	5	4
Familiarity with UIs for divergent and convergent thinking	–	–	–	5	4
Familiarity with ITs for divergent and convergent thinking	–	–	–	4	4

The 5 recruited participants comprised 3/5 OE experts (P1, P2, P3) and 2/5 HCI researchers (P4, P5), with 2 female and 3 male participants, experience ranging from 5 to 9 years, and academic levels spanning PhD candidates and postdoctoral researchers. Self-reported familiarity scores confirmed relevant domain expertise across all recruited participants: OE participants (P1, P2, P3) reported averaged scores of 5/5 for requirements elicitation in OE, 4.33/5 for the use of CQs for requirements elicitation in OE, and 3.67/5 for the use of LLMs for requirements elicitation with CQs in OE; HCI participants (P4, P5) reported averaged scores of 4.5/5 for divergent and convergent thinking, 4.5/5 for UIs supporting divergent and convergent thinking, and 4/5 for ITs supporting divergent and convergent thinking, with all individual scores at or above 3/5 (Table 5).

We further acknowledge that single-institution recruitment may introduce homogeneity bias and that 7 participants (including A1 and A2) are a small sample. However, both are acceptable: (1) participants bring diverse experience levels accumulated across prior and current research projects, with each participant engaged in a distinct research project at the time of the workshop, ranging from 5 to 9 years of experience across OE and HCI, and varying familiarity levels (3–5 on a 5-point Likert scale) with the workshop topics, as evidenced in Table 5, reducing the risk of perspective homogeneity; and (2) for a design study whose goal is to produce a theoretically grounded interaction model through expert deliberation, implement it as a working prototype, and explore its applicability through a practical use case, a small focused expert sample is appropriate, with all outputs serving as a basis for future empirical evaluation with a broader and independent participant population.

4.2. *Session 1: empathise, define, ideate, and prototype*

The Empathise stage lasted approximately 20 minutes. OE experts (A1, P1, P2) presented their current ontology requirements elicitation workflows and the challenges they face in LLM-based ontology scoping with existing tools, structured around two prompts: what challenges they currently face in ontology scoping, and where existing tools fail to support their work. HCI participants (A2, P4, P5) listened and asked clarifying questions to build sufficient understanding of OE practice to inform the subsequent design stages. This ensured that the problem space was defined by the actual challenges of OE practice rather than by HCI assumptions about what those challenges might be. The Define stage lasted approximately 20 minutes. All participants collaboratively synthesised the challenges from the Empathise stage into a set of focused problem statements. Three problem statements were identified and agreed upon by all participants, as reported in Table 6.

Table 6
Problem statements produced in the Define stage

ID	Problem statement	Description
PS1	Current automated LLM-based CQ generation tools provide little auditing support	Current LLM-based CQ generation tools such as AgOCQs [12], NeOn-GPT [53], RevOnt [40], and RETROFIT-CQs [5] are designed for generation only and provide little support for ontology engineers to subsequently audit whether generated CQ candidates are complete and relevant to define the target ontology scope.
PS2	Current LLM-based conversational agents cause fixation through linear interaction	Conversational agents such as OntoChat [180, 185] (Fig. 5) support auditing through follow-up questions and varied suggestions, but their turn-based chatbot interfaces are inherently linear and may cause fixation [144, 145], where ontology engineers become anchored to their initial inputs and fail to adequately explore the full ontology scope.
PS3	Current ontology editors and mind-mapping tools lack dimension-oriented organisation for CQ-based scope auditing	Ontology editors such as Protégé [109] and metaphactory [69] provide two UIs: nested list-based UIs (Fig. 2) that organise concepts and relations hierarchically, requiring constant scrolling and making it difficult to maintain a global overview; and graph-based UIs of the ontology structure itself, which can provide a global overview but are not organised along the dimensions of subdomains and term granularity, making it difficult to reason about whether CQ candidates collectively achieve the intended ontology scope. General mind-mapping tools such as Miro[3] and XMind[4] offer flexible UIs for personalising the scope auditing process but are underexplored for ontology scoping purposes.

The Ideate stage lasted approximately 30 minutes and was structured in two phases corresponding to divergent and convergent thinking. In the divergent phase, A1 and A2 presented the ITs and UIs identified in Section 3.5, and all participants collectively discussed how each problem statement in Table 6 could be addressed through the application of specific ITs and UIs. HCI researchers also presented a range of interaction design frameworks that held potential as organising structures for mapping the identified ITs and UIs into a coherent interaction model for supporting divergent and convergent thinking, including Norman's action cycle [114], the activity theory framework [84], and Shneiderman's visual information-seeking mantra [136].

In the convergent phase, participants discussed and converged on Shneiderman's visual information-seeking mantra [136] as the organising structure for two reasons: first, its three-stage sequence of overview first, zoom and filter, and details on demand directly maps onto the progression from divergent to convergent thinking in ontology scoping, where ontology engineers must first establish a global overview of the CQ space before narrowing attention to specific regions and then inspecting individual regions or CQ candidates in detail; and second, it is grounded in spatial navigation and visual reasoning, providing a principled basis for addressing PS3. All decisions were reached through discussion to consensus. The selected ITs and UIs and their relationships to the problem statements are reported in Table 7.

Table 7

The conceptual interaction model produced in the Ideate stage, comprising the selected ITs and UIs mapped onto the three stages of Shneiderman's visual information-seeking mantra [136], the problem statements they address, and the divergent and/or convergent thinking they support.

No.	UI / IT	Addresses	Thinking mode	Rationale
Overview first				
UI-2	Canvas	PS1, PS2, PS3	Divergent + Convergent	Places LLM-generated CQ candidates on a two-dimensional grid along subdomains and term granularity, letting ontology engineers see at a glance where the ontology scope has gaps or too many candidates.
UI-11	Nodes	PS1, PS3	Divergent + Convergent	Shows each CQ candidate as a node on the grid, making it easy to inspect and compare individual candidates across subdomains and term granularity levels.
IT-6	Clustering	PS1, PS3	Divergent + Convergent	LLM-based system groups CQ candidates with shared subdomains or term granularity into clusters, showing where candidates are concentrated or sparse across the ontology scope.
IT-7	Click-to-expand	PS1, PS3	Convergent	Opens a CQ node to show the LLM-suggested terms for that candidate's subdomain and granularity level, letting ontology engineers check, refine, and eliminate terms at the detail level without losing the overall view.
Zoom and filter				
IT-1	Spatial navigation	PS1, PS2, PS3	Divergent + Convergent	This lets ontology engineers move freely across clusters, subdomains, and granularity levels without following a fixed order, directly addressing fixation caused by linear interaction.
UI-10	In-flow options	PS1, PS2	Divergent	LLM-based system shows inline suggestions for new CQ candidates, terms, or dimension values at the point of editing, letting ontology engineers expand the ontology scope without stopping their current work.
Details on demand				
IT-2	Zooming	PS1, PS2, PS3	Divergent + Convergent	Switches between a full view of the ontology scope and a close-up of a specific subdomain or granularity level, supporting both broad assessment and detailed inspection of CQ candidates.
UI-4	Focus region in context	PS1, PS2, PS3	Convergent	Displays a selected region of the ontology scope in detail alongside the full scope overview; the LLM-based system generates CQ and term candidates within the selected region based on the surrounding context, keeping local suggestions consistent with the overall ontology scope.
UI-7	Binary rating	PS1	Convergent	This lets ontology engineers delete CQ candidates, terms, or dimension values as irrelevant; the LLM uses these deletions to stop similar suggestions again, progressively narrowing the ontology scope.

The Prototype stage lasted approximately 50 minutes. Based on the conceptual interaction model produced in the Ideate stage (Table 7), all participants collaboratively produced a visual prototype illustrating how the selected ITs and UIs work together to support LLM-based ontology scoping with CQs, as illustrated in Fig. 11, using Miro[14] as a shared collaborative workspace. This visual prototype was subsequently used by A1 and A2 as the direct specification for implementing OntoScope after this session.

[14]https://miro.com

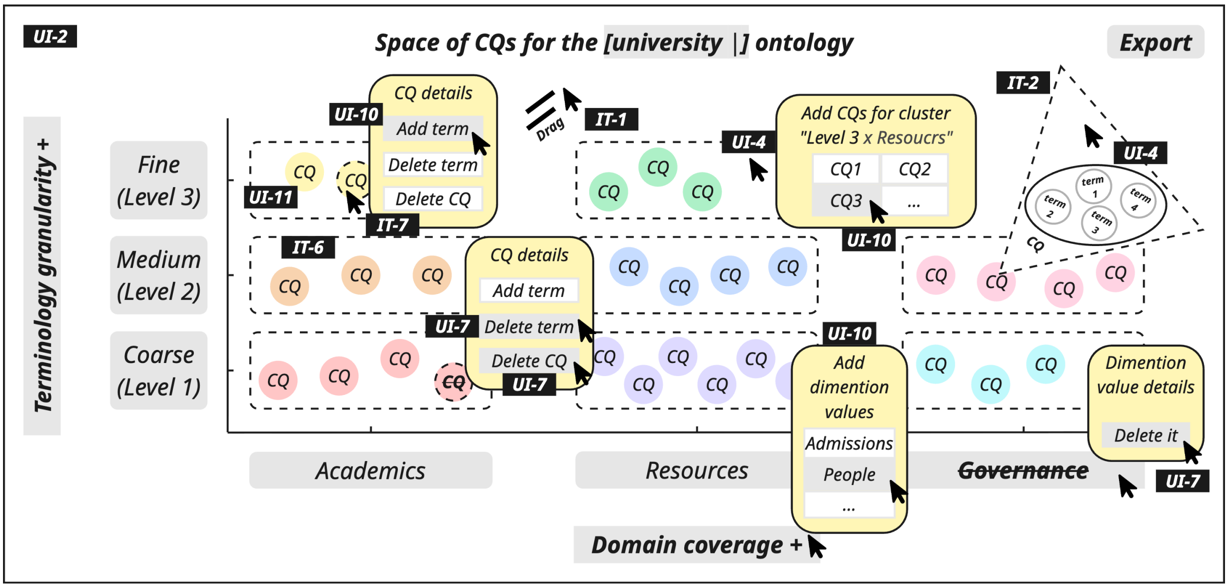


Fig. 11. The visual prototype produced in the Prototype stage, showing how the selected ITs and UIs work together to support LLM-based ontology scoping with CQs along the dimensions of subdomains (horizontal) and term granularity (vertical). The prototype illustrates the spatial overview of LLM-generated CQ candidates as manipulable nodes (UI-11) organised on a canvas (UI-2), grouped into clusters (IT-6), with navigation across regions (IT-1), zooming between global and detailed views (IT-2), in-flow suggestions for new CQ candidates, terms, and dimension values (UI-10), focus region in context for local refinement (UI-4), click-to-expand for term-level detail (IT-7), and binary rating for deleting irrelevant CQ candidates, terms, or dimension values (UI-7).

4.3. OntoScope

OntoScope is a web-based system prototype implementing the conceptual interaction model (Table 7) produced in the Ideate stage, with the visual prototype (Fig. 11) serving as its design specification. It supports ontology engineers in scoping an ontology by organising LLM-generated CQ candidates spatially along two dimensions: subdomains and term granularity. The frontend is implemented in React with Vite, with Tailwind CSS and Shadcn/ui for accessible and responsive interface design, and D3.js for interactive spatial visualisation of the two-dimensional CQ candidate space, selected for its support for custom data-driven spatial layouts not available in standard charting libraries. The backend is implemented in Node.js and Express.js, providing a RESTful API that handles real-time LLM requests for CQ, term, and dimension value generation, decoupling the frontend interaction layer from the LLM backend to allow each to be updated independently.

OntoScope uses OpenAI's `gpt-4.1`[15] as its underlying language model, selected over `gemini-2.5-pro`[16] and `claude-opus-4-20250514`[17] through pilot testing by the first and second authors, who independently evaluated and consistently agreed that `gpt-4.1` produced CQ candidates with higher well-formedness, terminological precision, and relevance to the target ontology scope across multiple test domains under the same task conditions. Pilot testing also identified that CQ generation prompts must instruct the model to treat singular and plural forms of terms as distinct, since they may correspond to different answers and therefore different ontological commitments. To remove redundant suggestions, OntoScope employs an automatic regeneration pipeline that checks for duplicates both within a newly generated set and against content already present in the ontology scope, capped at ten iterations, beyond which pilot testing showed that additional iterations yield negligible new valid suggestions while substantially increasing response latency.

[15]https://openai.com/index/gpt-4-1/
[16]https://modelcards.withgoogle.com/assets/documents/gemini-2.5-pro.pdf
[17]https://docs.claude.com/en/docs/about-claude/models/overview

4.4. Session 2: test

The Test session lasted approximately 1 hour and was attended by A1, A2, and P3. P3, an OE expert with 5 years of experience who had not attended the first session, served as an evaluator unfamiliar with the design decisions made during the first session, providing an evaluation perspective unaffected by involvement in the design process. In the first 15 minutes, A1 introduced the background of the work and clarified the task: P3 was asked to use OntoScope to scope a university ontology as completely and relevantly as possible within 30 minutes, a domain chosen because it is broadly familiar and requires no specialist prior knowledge.

A1 then walked through OntoScope with P3 for 15 minutes, answering any clarification questions about the system before the evaluation began. The session was screen- and audio-recorded, and P3 was asked to think aloud throughout the interaction; think-aloud [184, 185] was selected because it enables participants to verbalise their reasoning, observations, and difficulties as they occur, providing direct evidence of whether the interaction model addresses the problem statements identified in the Define stage (Table 6), without imposing a predefined response structure on the evaluator. A2 documented observations and feedback in real time.

This session constitutes a use case demonstration [61, 170]: its goal was not to produce statistically generalisable results or to evaluate against a benchmark, but to explore through a realistic use case how OntoScope can potentially support an ontology engineer's divergent and convergent thinking in LLM-based ontology scoping with CQs, and how this support can potentially contribute to a more complete and relevant ontology scope in practice, which is a validated and widely adopted approach for evaluating early-stage interactive system prototypes [61, 170].

4.5. Use case demonstration

This section demonstrates how OntoScope can potentially support an ontology engineer in scoping a university ontology, using P3's think-aloud session as the practical use case. Fig. 12 shows the main interface of OntoScope, and Fig. 13 shows the three popup windows; both are referenced throughout the demonstration. The demonstration follows P3's workflow sequentially, showing how the interaction model for supporting divergent and convergent thinking addresses the three problem statements identified in the Define stage (Table 6) and how this support can potentially contribute to a more complete and relevant ontology scope.

P3 entered `university` in the domain input field, and OntoScope generated an initial CQ space (UI-2, Fig. 12) with three subdomains (`academics`, `people`, and `facilities`) crossed with three granularity levels, each cell populated with LLM-generated terms, each associated with one CQ candidate. Unlike generation-only tools such as AgOCQs [12] and NeOn-GPT [53] which present candidates as a flat list, and unlike ontology editors such as Protégé [109] and mind-mapping tools such as Miro[3] which do not organise content along the dimensions of subdomains and term granularity, the canvas places all candidates spatially along the two dimensions required for CQ-based scope auditing (addressing PS1 and PS3), giving P3 a basis for reasoning about dimensional coverage and potentially supporting divergent thinking before committing to any specific region.

P3 zoomed out (IT-2, Fig. 12) to assess the global distribution of candidates and noticed that subdomains such as `research` and `services` were absent. The spatial overview provided a basis for this observation, suggesting that the scope may have been too narrow at the subdomain level before any CQ candidates had been added, a form of auditing that generation-only tools do not support (addressing PS1). This observation prompted divergent thinking: P3 clicked the subdomain axis to open Popup Window 1 (UI-10, Fig. 13), where the LLM suggested missing subdomains based on the current dimensional distribution. P3 entered custom subdomains and selected relevant suggestions until they became loosely related to the intended scope. As P3 noted: *"It feels like I can just keep opening up new subdomains… (after adding a while) OK, looks like they are all related, and I can't believe I'm gonna miss so many things before."*

With the subdomains expanded, P3 populated CQ candidates across the grid, navigating freely between cells by dragging (IT-1, Fig. 12). Unlike conversational agents such as OntoChat [180], where generation order may constrain review order and potentially cause fixation, the spatial canvas allowed P3 to choose which cell to populate next based on what the overview suggested was most in need of attention (UI-4) (addressing PS2). He clicked the `academics` × `First-level` cell to open Popup Window 2 (UI-10, Fig. 13), where the LLM generated contextually relevant CQ candidates across the three CQ patterns aligned with that specific cell. P3 entered custom

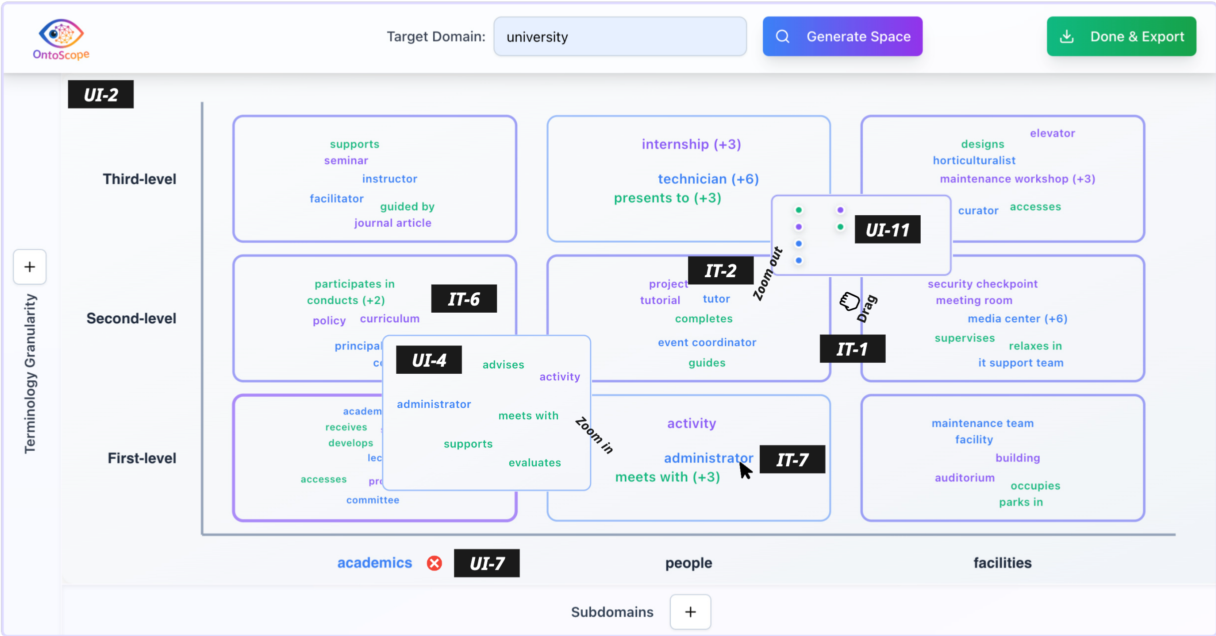


Fig. 12. The main interface of OntoScope, showing the two-dimensional canvas (UI-2) organising LLM-generated CQ candidates along subdomains (horizontal) and term granularity levels (vertical). The interface supports spatial navigation by dragging (IT-1), zooming between global and detailed views (IT-2), LLM-based clustering of CQ candidates (IT-6), focus region in context for local refinement alongside the global overview (UI-4), click-to-expand for CQ candidate details (IT-7), and deletion of irrelevant subdomains and granularity levels (UI-7).

CQs and selected relevant suggestions until they matched existing candidates. As P3 noted: *"Without it (referring to OntoScope) I would have struggled to define the domain properly, since each recommendation reminded me of something I had overlooked before."*

After populating all cells, P3 added a fourth granularity level to test whether finer granularity was needed (Fig. 12). Seeing the new row appear across all subdomains simultaneously provided a basis for a convergent decision: P3 judged the new candidates too detailed across the entire scope and deleted the entire level using the delete control (UI-7), a cross-dimensional assessment that may be more difficult to make in ontology editors and mind-mapping tools (addressing PS3). As P3 noted: *"I don't think a fourth level is needed... (after thinking for a while) looks like they are too detailed... yeah (pointing to the third granularity level), if you look at level three, it is quite good there."*

P3 then refined term-level detail by clicking individual terms on the canvas (IT-7, Fig. 12) to open Popup Window 3 (Fig. 13). For example, clicking a term in the `academics × First-level` cell opened the CQ details for "What does academic staff coordinate?", showing existing terms `program`, `course`, `event`, and `workshop`, alongside LLM suggestions `committee`, `project`, and `initiative`. Reviewing the full term set prompted convergent thinking: P3 noticed that `event` and `workshop` were semantically similar and deleted `workshop` using the delete control (UI-7), after which the LLM suppressed similar terms in subsequent cycles. The LLM suggestions `committee` and `project` introduced new ontological distinctions, prompting a divergent decision to accept them (UI-10), addressing PS1 by enabling term-level auditing that generation-only tools do not support. As P3 noted: *"Ah! the terms it suggested always felt more spot-on than what I would have come up with myself... it's like it put into words exactly what I was trying to say but couldn't quite get there."*

Finally, P3 zoomed out (IT-2, Fig. 12) to reassess the full scope. The global overview provided a basis for a final round of convergent decisions: P3 judged the `governance` subdomain, which had been manually added by P3 in a previous step, redundant relative to the rest of the scope and deleted it (by using UI-7, Fig. 12). P3 also removed semantically similar CQ candidates and terms and refined unsatisfied CQs until satisfied (by using UI-7, Fig. 13). As P3 noted: *"I really like the feeling of having the full picture in front of me... it just naturally drew*

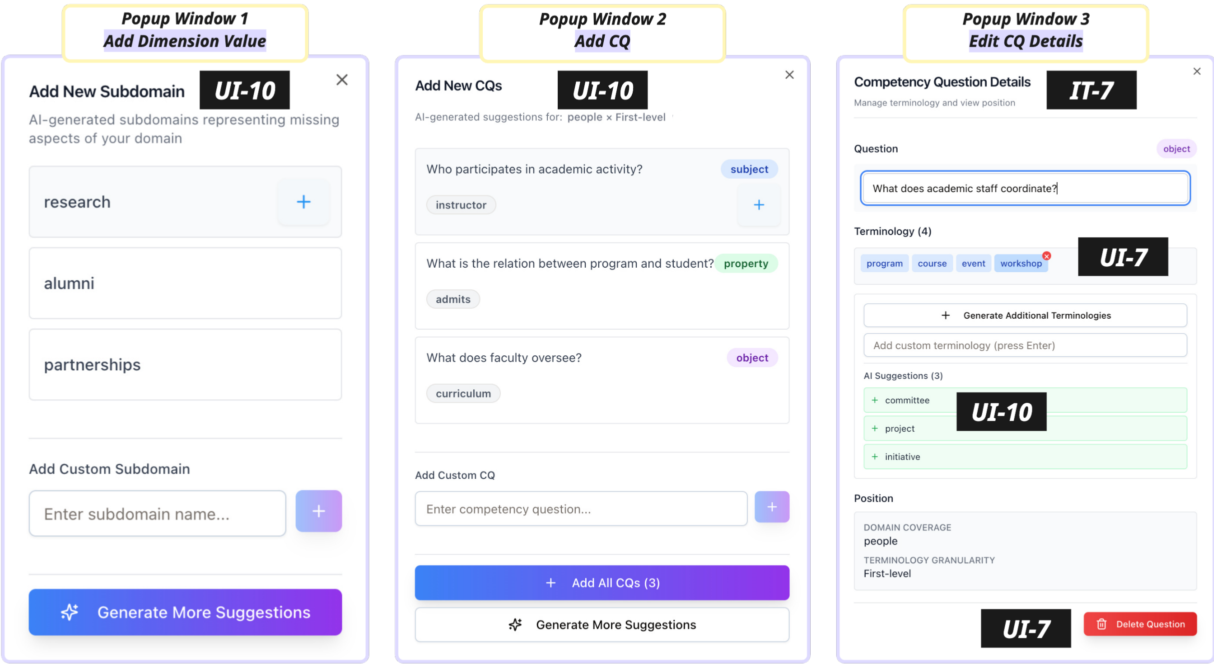


Fig. 13. Three popup windows in OntoScope, all constituting in-flow options (UI-10): Popup Window 1 for adding new subdomains with LLM-generated suggestions based on the current scope; Popup Window 2 for adding new CQ candidates within a specific subdomain and granularity level across the three CQ patterns; and Popup Window 3 for editing CQ candidate details including LLM-suggested terms and deletion of irrelevant terms or CQ candidates (UI-7).

my attention to where things needed fixing, without making me feel overwhelmed." After P3 was satisfied with the scope, he exported the finalised ontology scope as a JSON file for downstream OE tasks and reflected on the overall experience: *"It (referring to OntoScope) felt like I was actually in control... I went through everything it suggested, and if something didn't feel right I could just change it. It wasn't as tiring as I expected."* This workflow suggests that OntoScope's interaction model may support fluid transitions between divergent and convergent thinking across all stages of LLM-based ontology scoping with CQs, with the potential to contribute to a more complete and relevant ontology scope than existing generation-only tools, conversational agents, or ontology editors can achieve independently.

5. Discussion

This section discusses four aspects of the work. Section 5.1 addresses concerns about LLM reliability in ontology scoping and how OntoScope's interaction model embeds ontology engineer auditing at every stage where LLM-generated content enters the ontology scope. Section 5.2 discusses the transferability of the identified interaction design findings to ontology requirements elicitation, grounding this transfer in the shared cognitive task structure between arts and creativity systems and LLM-based ontology scoping with CQs. Section 5.3 discusses the broader implications of the identified ITs and UIs for OE and the Semantic Web community, including their potential applicability to other OE tasks and their potential for supporting domain experts and end users without prior OE expertise. Section 5.4 identifies the evaluation directions needed to move beyond the current use case demonstration, including a systematic user evaluation of OntoScope with a broader and independent participant population and a technical evaluation of LLM output quality against established ontology benchmarks.

5.1. *Addressing LLM reliability concerns in ontology scoping*

A concern when using LLM-generated content in OE tasks is that LLMs may produce plausible-sounding but semantically hollow, factually incorrect, or contextually irrelevant suggestions, and it is not always straightforward for an ontology engineer to identify such content [96, 129, 182]. OntoScope's interaction model can potentially mitigate this concern by embedding auditing at every stage where new LLM-generated content enters the ontology scope. When the ontology engineer clicks a cell to add new CQ candidates (Popup Window 2, Fig. 13), when they click the subdomain axis to add new dimension values (Popup Window 1, Fig. 13), and when they click a term to inspect and extend its associated CQ (Popup Window 3, Fig. 13), the LLM generates suggestions that the ontology engineer must explicitly accept or reject before they are incorporated into the ontology scope, meaning no content is added without a deliberate decision.

At a higher level, the spatial overview (UI-2, Fig. 12) supports auditing across the full ontology scope by making the full set of accepted content visible across both dimensions simultaneously, enabling the ontology engineer to reason about the collective distribution of candidates across subdomains and granularity levels rather than evaluating each candidate in isolation. This dimensional reasoning may surface issues that individual inspection alone would not reveal: for instance, a region that appears sparse or dense relative to others may indicate that the LLM has under- or over-generated in a particular part of the dimensional space, prompting further review. The binary rating interaction (UI-7, Fig. 13) allows the ontology engineer to delete any content judged irrelevant or incorrect at any point, after which the LLM suppresses similar suggestions in subsequent generation cycles. However, whether this combination of interactions is sufficient to mitigate the propagation of hallucinated content into the final ontology scope remains an empirical question, and a more systematic evaluation of LLM output quality within OntoScope against established ontology benchmarks is an important direction for future work.

5.2. *Transferability of interaction design findings to ontology requirements elicitation*

Ontology requirements elicitation is the process in which ontology engineers define the scope of concepts and relations an ontology must cover to adequately serve its intended application domain [7, 46]. CQs operationalise this process by expressing these requirements as natural language questions that the ontology must be able to answer [63, 115]. Ontology scoping, the stage in which ontology engineers determine which CQ candidates are complete and relevant to define the target ontology scope, is where this process is most cognitively demanding: the intended scope cannot be fully anticipated in advance, requiring ontology engineers to move iteratively between divergently exploring and generating CQ candidates and convergently evaluating and refining them until a well-defined scope is reached [46, 183]. The arts and creativity domain has studied this kind of cognitive challenge extensively. A substantial body of LLM-based interactive systems has been developed to support divergent and convergent thinking in creative tasks [10, 145], and the ITs and UIs identified in this survey are drawn from these systems. Their potential applicability to ontology requirements elicitation is grounded in the shared cognitive task structure: in both contexts, users work toward a goal whose full scope is not known in advance and must move iteratively between expanding and narrowing a candidate space organised along task-relevant dimensions.

The transferability of these findings is further explored by the design thinking workshop and use case demonstration reported in this work. The workshop showed that OE experts and HCI researchers, when presented with the identified ITs and UIs, were able to collectively identify which of them were potentially applicable to LLM-based ontology scoping and justify how, producing a conceptual interaction model that maps directly onto the cognitive demands of the scoping task. The use case demonstration then illustrated that an ontology engineer could use OntoScope, which implements this model, to reason about the distribution of LLM-generated CQ candidates across subdomains and term granularity levels, identify gaps and overlaps, and make targeted auditing decisions, suggesting that the identified interaction designs may support similar divergent and convergent thinking processes in ontology scoping as they do in the arts and creativity domain. Together, these two outputs suggest that the transfer from arts and creativity to ontology requirements elicitation is theoretically justified by the shared cognitive task structure and potentially realisable in practice. The identified ITs and UIs can therefore potentially serve as a reference for ontology requirements elicitation tool developers, with different subsets combinable with different organising frameworks to compose interaction models suited to different OE contexts.

5.3. *Implications of interaction design findings for OE and the Semantic Web*

The 7 ITs and 14 UIs identified in this survey were drawn from systems built for supporting divergent and convergent thinking, but they reflect fundamental interaction design principles such as spatial reasoning, progressive disclosure, and direct manipulation that are not specific to any single domain [112, 136]. Researchers and tool developers working on other OE tasks or Semantic Web applications that involve human reasoning over LLM-generated content, such as user story elicitation [180, 185], KG population [119], or ontology alignment [52], face a similar challenge: large volumes of LLM-generated candidates must be audited by a human expert who needs to reason about coverage, identify gaps, and make targeted decisions. The identified interaction designs, which were specifically evaluated in this survey for how well they support this kind of reasoning, may therefore provide a useful starting point for tool developers working on these tasks.

Beyond task-specific interaction model design, the identified pool also holds potential for selection based on the background of the users involved. The OE community has increasingly advocated for tools accessible to domain experts, stakeholders, and end users rather than trained ontology engineers, with the aim of broadening participation in ontology building [46, 138]. Different user groups bring different interaction familiarity from their daily work, and selecting interaction designs that align with that familiarity may lower the learning barrier and make OE tasks more accessible. For example, historians or archivists who regularly work with chronological documentation may find it more natural to work with content organised along a temporal structure (UI-13) [181, 186], while music composers or audio engineers accustomed to multi-track production environments may prefer to work with content organised through a lane and track structure (UI-14) [158, 160].

The identified findings provide a pool from which user-appropriate interaction models can be composed, allowing domain experts to engage with these tasks through interaction designs that are consistent with their existing mental models and working practices [112]. This matters because domain experts bring irreplaceable knowledge about the subject matter, but may disengage or make suboptimal decisions when the interaction design is unfamiliar or cognitively demanding [117, 149]. By selecting interaction designs that reduce this mismatch, tool developers can potentially lower the cognitive overhead of the OE task itself, allowing domain experts to focus their attention on the content rather than on navigating an unfamiliar interface [46, 138].

5.4. *Towards systematic evaluation*

The conceptual interaction model (Table 7), implemented as OntoScope (Section 4.3), and the use case demonstration (Section 4.5) provide initial evidence that the interaction model can potentially support an ontology engineer in LLM-based ontology scoping with CQs in a realistic context, but cannot produce statistically generalisable results. A systematic user evaluation with a broader and independent participant population would be needed to assess whether the interaction model reliably supports divergent and convergent thinking across different ontology engineers and ontology domains, and whether this support contributes to a more complete and relevant ontology scope compared to existing tools. Such an evaluation could draw on the Creativity Support Index [36], which provides a validated and widely used instrument for measuring the degree to which an interactive system supports creative cognitive work across dimensions such as exploration, expressiveness, and results worth effort, making it applicable to the divergent and convergent thinking support that the interaction model aims to provide.

Alongside this, domain-specific measures of ontology scope quality would be needed to assess the outcome of the scoping process itself. For this purpose, RevOnt [40], which provides a benchmark of validated CQ sets across 20 representative domains on Wikidata, offers a potential ground truth against which the completeness and relevance of OntoScope-produced ontology scopes could be assessed. F1 score is a suitable metric for this purpose, as it jointly measures the proportion of ground truth CQs recovered by the system and the proportion of system-produced CQs that are relevant to the intended scope, providing a balanced measure of scope quality that accounts for both completeness and precision. A technical evaluation of LLM output quality within OntoScope using such metrics would further strengthen the evidence base for the system's reliability in practice, complementing the user evaluation and addressing the hallucination auditing concerns discussed in Section 5.1.

6. Limitations

This work has several limitations. Regarding the SLR, the corpus is constrained by the time of the search and the scope of the three selected databases, and given the rapid expansion of LLM-related publications, the SLR may not capture the full breadth of relevant systems. Additionally, the SLR focused specifically on interaction patterns that explicitly support divergent and convergent thinking in arts and creativity systems, which may have omitted ITs embedded in general-purpose LLM systems that were not framed under this cognitive model. Regarding the design thinking workshop, the study was conducted with 7 participants from a single institution, which may introduce homogeneity bias and limit the generalisability of the design decisions to broader OE communities. However, this is acceptable for a design study of this nature, where the goal is to produce a theoretically grounded interaction model through expert deliberation rather than to generalise findings across populations, and the diversity of expertise and experience levels across the participants, as evidenced in Table 5, reduces the risk of perspective homogeneity within the workshop itself. Regarding evaluation, OntoScope was evaluated through a single use case demonstration with one evaluator, which is intended to show that the interaction model is potentially applicable in a realistic context rather than to measure its effectiveness, and therefore cannot produce statistically generalisable results about the effectiveness of the interaction model across different ontology engineers, ontology domains, or LLM configurations; a systematic user evaluation with a broader and independent participant population and a technical evaluation against established ontology benchmarks remain important directions for future work.

7. Conclusions

This paper presented a survey on interaction design with LLMs for supporting divergent and convergent thinking in arts and creativity systems, identifying 7 ITs and 14 UIs, each justified with respect to how it supports these thinking modes, with IRR assessed for each identified category to ensure the stability of the findings. We then argued that these findings are transferable to ontology requirements elicitation with CQs, given the shared cognitive task structure: in both contexts, the target solution space is multi-dimensional, ill-defined, and cannot be fully anticipated in advance, requiring users to move between divergently exploring and generating candidates and convergently evaluating and refining them until a well-defined scope is reached.

To explore this transferability, we conducted a design thinking workshop with 7 participants from King's College London, producing a conceptual interaction model that maps a selected subset of the identified ITs and UIs onto the three stages of Shneiderman's visual information-seeking mantra for LLM-based ontology scoping with CQs, a system prototype implementing that model called OntoScope, and a use case demonstration showing how OntoScope can potentially support an ontology engineer in scoping a university ontology. The use case demonstration suggests that the interaction model can potentially support ontology engineers in identifying gaps and overlaps in the CQ candidate space, navigating it non-sequentially to prevent fixation, and auditing LLM-generated content at every stage where new content enters the ontology scope, contributing to a more complete and relevant ontology scope than existing generation-only tools, conversational agents, or ontology editors can achieve independently.

The identified findings can serve as a reference for ontology requirements elicitation tool developers selecting interaction designs for LLM-based systems, and the conceptual interaction model and OntoScope demonstrate one possible realisation of these findings, though many other interaction models combining different subsets of the identified techniques and UIs remain possible for different OE contexts. Future work includes a systematic user evaluation of OntoScope with a broader and independent participant population, a technical evaluation of LLM output quality against established ontology benchmarks, and an exploration of how the identified interaction designs can be applied to other OE and Semantic Web applications where similar challenges of human reasoning and auditing over LLM-generated content arise.

Acknowledgements

The authors acknowledge the use of Claude in editing language to support rephrasing, writing clarity, and grammar correction. All substantive ideas, arguments, design decisions, and research directions were developed by the authors. This work was supported by the Engineering and Physical Sciences Research Council [grant number EP/Y009800/1], funded through Responsible AI UK (KP0011); SIEMENS AG; and the Institute for Advanced Study, Technical University of Munich, Germany.

References

[1] K. Adamkiewicz, P.W. Woźniak, J. Dominiak, A. Romanowski, J. Karolus and S. Frolov, PromptMap: An Alternative Interaction Style for AI-Based Image Generation, in: *Proceedings of the 30th International Conference on Intelligent User Interfaces*, 2025, pp. 1162–1176.

[2] C. Ahlberg and B. Shneiderman, The alphaslider: a compact and rapid selector, in: *Proceedings of the SIGCHI conference on Human factors in computing systems*, 1994, pp. 365–371.

[3] C. Ahlberg, C. Williamson and B. Shneiderman, Dynamic queries for information exploration: An implementation and evaluation, in: *Proceedings of the SIGCHI conference on Human factors in computing systems*, 1992, pp. 619–626.

[4] R. Alharbi, Automating the Formulation of Competency Questions in Ontology Engineering, PhD thesis, University of Liverpool, 2025.

[5] R. Alharbi, V. Tamma, F. Grasso and T. Payne, An experiment in retrofitting competency questions for existing ontologies, in: *Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing*, ACM, 2024, pp. 1650–1658.

[6] R. Alharbi, V. Tamma, F. Grasso and T.R. Payne, Investigating Open Source LLMs to Retrofit Competency Questions in Ontology Engineering, in: *Proceedings of the AAAI Symposium Series*, Vol. 4, 2024, pp. 188–198.

[7] R. Alharbi, V. Tamma, F. Grasso and T.R. Payne, A review and comparison of competency question engineering approaches, in: *International Conference on Knowledge Engineering and Knowledge Management*, Springer, 2024, pp. 271–290.

[8] R. Alharbi, V. Tamma, F. Grasso and T.R. Payne, The role of Generative AI in competency question retrofitting, in: *European Semantic Web Conference*, Springer, 2024, pp. 3–13.

[9] S.G. Almeda, J. Zamfirescu-Pereira, K.W. Kim, P. Mani Rathnam and B. Hartmann, Prompting for discovery: Flexible sense-making for ai art-making with dreamsheets, in: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–17.

[10] T. Angert, M. Suzara, J. Han, C. Pondoc and H. Subramonyam, Spellburst: A node-based interface for exploratory creative coding with natural language prompts, in: *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, 2023, pp. 1–22.

[11] M.-J. Antia and C.M. Keet, Assessing and enhancing bottom-up CNL design for competency questions for ontologies, in: *Proceedings of the Seventh International Workshop on Controlled Natural Language (CNL 2020/21)*, 2021.

[12] M.-J. Antia and C.M. Keet, Automating the Generation of Competency Questions for Ontologies with AgOCQs, in: *Iberoamerican Knowledge Graphs and Semantic Web Conference*, Springer, 2023, pp. 213–227.

[13] F. Baader, *The description logic handbook: Theory, implementation and applications*, Cambridge university press, 2003.

[14] B. Bach, Z. Wang, M. Farinella, D. Murray-Rust and N. Henry Riche, Design patterns for data comics, in: *Proceedings of the 2018 chi conference on human factors in computing systems*, 2018, pp. 1–12.

[15] B.H. Banathy, *Designing social systems in a changing world*, Springer Science & Business Media, 2013.

[16] L. Bartram, A. Ho, J. Dill and F. Henigman, The continuous zoom: A constrained fisheye technique for viewing and navigating large information spaces, in: *Proceedings of the 8th annual ACM symposium on User interface and software technology*, 1995, pp. 207–215.

[17] A. Barua, K. Benharrak, M. Chen, M. Huh and A. Pavel, Lotus: Creating Short Videos From Long Videos With Abstractive and Extractive Summarization, in: *Proceedings of the 30th International Conference on Intelligent User Interfaces*, 2025, pp. 967–981.

[18] M. Beaudouin-Lafon and W.E. Mackay, Prototyping tools and techniques, in: *The human-computer interaction handbook*, CRC Press, 2007, pp. 1043–1066.

[19] B.B. Bederson and J.D. Hollan, Pad++ a zooming graphical interface for exploring alternate interface physics, in: *Proceedings of the 7th annual ACM symposium on User interface software and technology*, 1994, pp. 17–26.

[20] A. Bezerianos, F. Chevalier, P. Dragicevic, N. Elmqvist and J.-D. Fekete, Graphdice: A system for exploring multivariate social networks, in: *Computer graphics forum*, Vol. 29, Wiley Online Library, 2010, pp. 863–872.

[21] C. Bezerra, Verifying description logic ontologies based on competency questions and unit testing (2017).

[22] C. Bezerra, F. Santana and F.L.G. de Freitas, CQChecker: A Tool to Check Ontologies in OWL-DL using Competency Questions written in Controlled Natural Language, *Learning and Nonlinear Models* **12** (2014), 115–129. https://api.semanticscholar.org/CorpusID:55282107.

[23] M.M. Biskjaer, P. Dalsgaard and K. Halskov, A constraint-based understanding of design spaces, in: *Proceedings of the 2014 conference on Designing interactive systems*, 2014, pp. 453–462.

[24] J.O. Borchers, A pattern approach to interaction design, in: *Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques*, 2000, pp. 369–378.

[25] I.d.V. Bosman, A.E. Smith, Y.L. Wong, K.S.D. Ka, D. Alemneh and A. Chow, Immersive technology in education, *Proceedings of the Association for Information Science and Technology* **61**(1) (2024), 721–724.

[26] N. Boukhelifa, A. Bezerianos, I.C. Trelea, N.M. Perrot and E. Lutton, An exploratory study on visual exploration of model simulations by multiple types of experts, in: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019, pp. 1–14.

[27] S. Brade, B. Wang, M. Sousa, S. Oore and T. Grossman, Promptify: Text-to-image generation through interactive prompt exploration with large language models, in: *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, 2023, pp. 1–14.

[28] A. Braham, F. Buendía, M. Khemaja and F. Gargouri, User interface design patterns and ontology models for adaptive mobile applications, *Personal and Ubiquitous Computing* **26**(6) (2022), 1395–1411. doi:10.1007/s00779-020-01481-5.

[29] V. Braun and V. Clarke, Using thematic analysis in psychology, *Qualitative research in psychology* **3**(2) (2006), 77–101.
[30] F. Brudy, C. Holz, R. Rädle, C.-J. Wu, S. Houben, C.N. Klokmose and N. Marquardt, Cross-device taxonomy: Survey, opportunities and challenges of interactions spanning across multiple devices, in: *Proceedings of the 2019 chi conference on human factors in computing systems*, 2019, pp. 1–28.
[31] A. Cai, S.R. Rick, J.L. Heyman, Y. Zhang, A. Filipowicz, M. Hong, M. Klenk and T. Malone, DesignAID: Using generative AI and semantic diversity for design inspiration, in: *Proceedings of The ACM Collective Intelligence Conference*, 2023, pp. 1–11.
[32] T. Chakrabarty, V. Padmakumar and H. He, Help me write a poem: Instruction tuning as a vehicle for collaborative poetry writing, *arXiv preprint arXiv:2210.13669* (2022).
[33] T. Chakrabarty, A. Saakyan, O. Winn, A. Panagopoulou, Y. Yang, M. Apidianaki and S. Muresan, I spy a metaphor: Large language models and diffusion models co-create visual metaphors, *arXiv preprint arXiv:2305.14724* (2023).
[34] L. Chen, Q. Jing, Y. Tsang, Q. Wang, R. Liu, D. Xia, Y. Zhou and L. Sun, AutoSpark: Supporting Automobile Appearance Design Ideation with Kansei Engineering and Generative AI, in: *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, 2024, pp. 1–19.
[35] M. Chen, A. Yang, S. Min, K.A. Hamilton and E. Wall, A novel lens on metacognition in visualization, in: *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, 2025, pp. 1–16.
[36] E. Cherry and C. Latulipe, Quantifying the creativity support of digital tools through the creativity support index, *ACM Transactions on Computer-Human Interaction (TOCHI)* **21**(4) (2014), 1–25.
[37] D. Choi, S. Hong, J. Park, J.J.Y. Chung and J. Kim, CreativeConnect: Supporting Reference Recombination for Graphic Design Ideation with Generative AI, in: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–25.
[38] J.J.Y. Chung and M. Kreminski, Patchview: LLM-Powered Worldbuilding with Generative Dust and Magnet Visualization, in: *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, 2024, pp. 1–19.
[39] J.J.Y. Chung, W. Kim, K.M. Yoo, H. Lee, E. Adar and M. Chang, TaleBrush: Sketching stories with generative pretrained language models, in: *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, 2022, pp. 1–19.
[40] F. Ciroku, J. de Berardinis, J. Kim, A. Meroño-Peñuela, V. Presutti and E. Simperl, RevOnt: Reverse engineering of competency questions from knowledge graphs via language models, *Journal of Web Semantics* (2024), 100822.
[41] J.H. Claessen and J.J. Van Wijk, Flexible linked axes for multivariate data visualization, *IEEE Transactions on Visualization and Computer Graphics* **17**(12) (2011), 2310–2316.
[42] N.J. Cooke, Knowledge elicitation, *Handbook of applied cognition* (1999), 479–509.
[43] N. Cross, Expertise in design: an overview, *Design studies* **25**(5) (2004), 427–441.
[44] J. de Berardinis, V.A. Carriero, N. Jain, N. Lazzari, A. Meroño-Peñuela, A. Poltronieri and V. Presutti, The polifonia ontology network: Building a semantic backbone for musical heritage, in: *International Semantic Web Conference*, Springer, 2023, pp. 302–322.
[45] E. De Bono and E. Zimbalist, *Lateral thinking*, Penguin London, 1970.
[46] A. De Nicola and M. Missikoff, A lightweight methodology for rapid ontology engineering, *Communications of the ACM* **59**(3) (2016), 79–86.
[47] Design Council, The Double Diamond: A Universally Accepted Depiction of the Design Process, Design Council, London, UK, 2005. https://www.designcouncil.org.uk/our-resources/the-double-diamond/.
[48] Design Council, The Double Diamond: A universally accepted depiction of the design process, 2005, Accessed: April 28, 2025. https://www.designcouncil.org.uk/our-resources/the-double-diamond/.
[49] A. Dix, *Human-computer interaction*, Vol. 1, Pearson Education, 2004.
[50] S.P. Dow, A. Glassco, J. Kass, M. Schwarz, D.L. Schwartz and S.R. Klemmer, Parallel prototyping leads to better design results, more divergence, and increased self-efficacy, *ACM Transactions on Computer-Human Interaction (TOCHI)* **17**(4) (2010), 1–24.
[51] A. Eshghi, I. Shalyminov and O. Lemon, Interactional dynamics and the emergence of language games., in: *FADLI@ ESSLLI*, 2017, pp. 17–21.
[52] J. Euzenat and P. Shvaiko, *Ontology matching*, Springer, 2007.
[53] N. Fathallah, A. Das, S.D. Giorgis, A. Poltronieri, P. Haase and L. Kovriguina, Neon-GPT: a large language model-powered pipeline for ontology learning, in: *European Semantic Web Conference*, Springer, 2024, pp. 36–50.
[54] J. Fereday and E. Muir-Cochrane, Demonstrating rigor using thematic analysis: A hybrid approach of inductive and deductive coding and theme development, *International journal of qualitative methods* **5**(1) (2006), 80–92.
[55] J.L. Fleiss and J. Cohen, The Equivalence of Weighted Kappa and the Intraclass Correlation Coefficient as Measures of Reliability, *Educational and Psychological Measurement* **33**(3) (1973), 613–619. doi:10.1177/001316447303300309.
[56] J. Frich, M. Nouwens, K. Halskov and P. Dalsgaard, How digital tools impact convergent and divergent thinking in design ideation, in: *Proceedings of the 2021 CHI conference on human factors in computing systems*, 2021, pp. 1–11.
[57] G.W. Furnas, Generalized fisheye views, *Acm Sigchi Bulletin* **17**(4) (1986), 16–23.
[58] M. Ghajargar, J. Bardzell and L. Lagerkvist, A redhead walks into a bar: experiences of writing fiction with artificial intelligence, in: *Proceedings of the 25th international academic MindTrek conference*, 2022, pp. 230–241.
[59] G. Gibbs, Analyzing Qualitative Data, *(No Title)* (2007).
[60] A. Gómez-Pérez, M. Fernández-López and O. Corcho, *Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*, Springer, 2004.
[61] S. Greenberg and B. Buxton, Usability evaluation considered harmful (some of the time), in: *Proceedings of the SIGCHI conference on Human factors in computing systems*, 2008, pp. 111–120.
[62] S. Greenberg, S. Carpendale, N. Marquardt and B. Buxton, *Sketching user experiences: The workbook*, Elsevier, 2012.

[63] M. Grüninger and M.S. Fox, The role of competency questions in enterprise engineering, in: *Benchmarking—Theory and practice*, Springer, 1995, pp. 22–31.

[64] N. Guarino, *Formal ontology in information systems: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy*, Vol. 46, IOS press, 1998.

[65] J.P. Guilford, Three faces of intellect. (1961).

[66] J.P. Guilford, The nature of human intelligence. (1967).

[67] Y. Guo, H. Shao, C. Liu, K. Xu and X. Yuan, Prompthis: Visualizing the process and influence of prompt editing during text-to-image creation, *IEEE Transactions on Visualization and Computer Graphics* (2024).

[68] D. Gustafsson, Analysing the Double diamond design process through research & implementation (2019).

[69] P. Haase, D.M. Herzig, A. Kozlov, A. Nikolov and J. Trame, metaphactory: A platform for knowledge graph management, *Semantic Web* **10**(6) (2019), 1109–1125.

[70] K. Halskov and C. Lundqvist, Filtering and informing the design space: Towards design-space thinking, *ACM Transactions on Computer-Human Interaction (TOCHI)* **28**(1) (2021), 1–28.

[71] S.G. Hart and L.E. Staveland, Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research, in: *Advances in psychology*, Vol. 52, Elsevier, 1988, pp. 139–183.

[72] E. Hauck and C. Aranha, Automatic generation of Super Mario levels via graph grammars, in: *2020 IEEE Conference on Games (CoG)*, IEEE, 2020, pp. 297–304.

[73] D. Hayatpur, D. Wigdor and H. Xia, Crosscode: Multi-level visualization of program execution, in: *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, 2023, pp. 1–13.

[74] Y. He, Language models for ontology engineering, PhD thesis, University of Oxford, Oxford, United Kingdom, 2024. doi:10.5287/ora-pd7q5y157.

[75] C. Heape, The Design Space: the design process as the construction, exploration and expansion of a conceptual space (2007).

[76] M.N. Hoque, T. Mashiat, B. Ghai, C.D. Shelton, F. Chevalier, K. Kraus and N. Elmqvist, The HaLLMark Effect: Supporting Provenance and Transparent Use of Large Language Models in Writing with Interactive Visualization, in: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–15.

[77] X. Hu, Y. Xing, X. Cai, Y. Zhao, M. Cook, R. Borgo and T. Neate, Designing interactions with generative AI for art and creativity: A systematic review and taxonomy, in: *Proceedings of the 2025 ACM Designing Interactive Systems Conference*, 2025, pp. 1126–1155.

[78] R. Huang, H. Lin, C. Chen, K. Zhang and W. Zeng, Plantography: Incorporating iterative design process into generative artificial intelligence for landscape rendering, in: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–19.

[79] M. Huh, Y.-H. Peng and A. Pavel, GenAssist: Making image generation accessible, in: *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, 2023, pp. 1–17.

[80] V.-T. Huynh, T.V. Nguyen and M.-T. Tran, Sketch2Reality: Immersive 3D indoor scene synthesis via sketches, in: *Proceedings of the 12th International Symposium on Information and Communication Technology*, 2023, pp. 863–869.

[81] R.J. Jacob, New human-computer interaction techniques, *NATO ASI SERIES F COMPUTER AND SYSTEMS SCIENCES* **129** (1994), 131–131.

[82] D.G. Jansson and S.M. Smith, Design fixation, *Design studies* **12**(1) (1991), 3–11.

[83] N. Jog and B. Shneiderman, Information visualization with smooth zooming on an starfield display, in: *Proc. IFIP Conf. Visual Databases*, Vol. 3, 1995, pp. 1–10.

[84] V. Kaptelinin and B.A. Nardi, *Acting with technology: Activity theory and interaction design*, MIT press, 2009.

[85] C.M. Keet, Z. Mahlaza and M.-J. Antia, CLaRO: a controlled language for authoring competency questions, in: *Research Conference on Metadata and Semantics Research*, Springer, 2019, pp. 3–15.

[86] E.A. Kemp, The role of the individual project in teaching knowledge acquisition, in: *Proceedings 1996 International Conference Software Engineering: Education and Practice*, IEEE, 1996, pp. 138–143.

[87] E.F. Kendall and D.L. McGuinness, *Ontology engineering*, Morgan & Claypool Publishers, 2019.

[88] J. Kim, S. Suh, L.B. Chilton and H. Xia, Metaphorian: Leveraging large language models to support extended metaphor creation for science writing, in: *Proceedings of the 2023 ACM Designing Interactive Systems Conference*, 2023, pp. 115–135.

[89] T.S. Kim, Y. Lee, M. Chang and J. Kim, Cells, generators, and lenses: Design framework for object-oriented interaction with large language models, in: *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, 2023, pp. 1–18.

[90] T. Kuhn, A Survey and Classification of Controlled Natural Languages, *Computational Linguistics* **40**(1) (2014), 121–170. doi:10.1162/COLI_a_00168. https://aclanthology.org/J14-1005/.

[91] T. Kuhn, A survey and classification of controlled natural languages, *Computational linguistics* **40**(1) (2014), 121–170.

[92] H. Kumar, J. Vincentius, E. Jordan and A. Anderson, Human creativity in the age of llms: Randomized experiments on divergent and convergent thinking, in: *Proceedings of the 2025 CHI conference on human factors in computing systems*, 2025, pp. 1–18.

[93] M. Lee, P. Liang and Q. Yang, Coauthor: Designing a human-ai collaborative writing dataset for exploring language model capabilities, in: *Proceedings of the 2022 CHI conference on human factors in computing systems*, 2022, pp. 1–19.

[94] Y. Li, R. García-Castro, N. Mihindukulasooriya, J. O'Donnell and S. Vega-Sánchez, Enhancing energy management at district and building levels via an EM-KPI ontology, *Automation in Construction* **99** (2019), 152–167.

[95] S. Lin, J. Warner, J. Zamfirescu-Pereira, M.G. Lee, S. Jain, S. Cai, P. Lertvittayakumjorn, M.X. Huang, S. Zhai, B. Hartmann et al., Rambler: Supporting Writing With Speech via LLM-Assisted Gist Manipulation, in: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–19.

[96] A.S. Lippolis, M.J. Saeedizade, R. Keskisärkkä, S. Zuppiroli, M. Ceriani, A. Gangemi, E. Blomqvist and A.G. Nuzzolese, Ontology generation using large language models, in: *European Semantic Web Conference*, Springer, 2025, pp. 321–341.

[97] V. Liu, Beyond text-to-image: Multimodal prompts to explore generative ai, in: *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*, 2023, pp. 1–6.

[98] V. Liu, J. Vermeulen, G. Fitzmaurice and J. Matejka, 3DALL-E: Integrating text-to-image AI in 3D design workflows, in: *Proceedings of the 2023 ACM designing interactive systems conference*, 2023, pp. 1955–1977.

[99] J.D. Lomas, M. Karac and M. Gielen, Design space cards: using a card deck to navigate the design space of interactive play, *Proceedings of the ACM on Human-Computer Interaction* **5**(CHI PLAY) (2021), 1–21.

[100] L. Long, C. Xinyi, W. Ruoyu, L. Toby Jia-Jun and L. Ray, Sketchar: Supporting character design and illustration prototyping using generative AI, *Proceedings of the ACM on Human-Computer Interaction* **8**(CHI PLAY) (2024), 337.

[101] A. MacEachren, D. Xiping, F. Hardisty, D. Guo and G. Lengerich, Exploring high-D spaces with multiform matrices and small multiples, in: *IEEE Symposium on Information Visualization 2003 (IEEE Cat. No. 03TH8714)*, IEEE, 2003, pp. 31–38.

[102] M. Masmoudi, S.B.A.B. Lamine, H.B. Zghal, M.H. Karray and B. Archimede, An ontology-based monitoring system for multi-source environmental observations, *Procedia Computer Science* **126** (2018), 1865–1874.

[103] D. Masson, S. Malacria, G. Casiez and D. Vogel, Directgpt: A direct manipulation interface to interact with large language models, in: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–16.

[104] M.L. McHugh, Interrater reliability: the kappa statistic, *Biochemia medica* **22**(3) (2012), 276–282.

[105] D. McKerlie and A. MacLean, Reasoning with design rationale: practical experience with design space analysis, *Design Studies* **15**(2) (1994), 214–226.

[106] G.A. Miller, WordNet: a lexical database for English, *Communications of the ACM* **38**(11) (1995), 39–41.

[107] G. Miriam Jacob, S. Sahu, C. Ito, M. Kawamura and B. Stenger, Text2Illustration: Sampling and Composing Creatives With Text, in: *Proceedings of the 7th Joint International Conference on Data Science & Management of Data (11th ACM IKDD CODS and 29th COMAD)*, 2024, pp. 538–541.

[108] P. Mirowski, K.W. Mathewson, J. Pittman and R. Evans, Co-writing screenplays and theatre scripts with language models: Evaluation by industry professionals, in: *Proceedings of the 2023 CHI conference on human factors in computing systems*, 2023, pp. 1–34.

[109] M.A. Musen, The protégé project: a look back and a look forward, *AI matters* **1**(4) (2015), 4–12.

[110] A. Muzahid, W. Wan, F. Sohel, N.U. Khan, O.D.C. Villagómez and H. Ullah, 3D object classification using a volumetric deep neural network: An efficient octree guided auxiliary learning approach, *IEEE Access* **8** (2020), 23802–23816.

[111] D. NORMAN, Some observations on mental models, *Mental Models* (1983), 7–14.

[112] D.A. Norman, Cognitive engineering, in: *User centered system design*, CRC Press, 1986, pp. 31–62.

[113] D.A. Norman, *The design of everyday things*, Basic Books, 2002.

[114] D.A. Norman and S.W. Draper, *User centered system design; new perspectives on human-computer interaction*, L. Erlbaum Associates Inc., 1986.

[115] N.F. Noy, Ontology Development 101: A Guide to Creating Your First Ontology, Technical Report, KSL-01-05, SMI-2001-0880, Stanford University, Knowledge Systems Laboratory, 2001. https://protege.stanford.edu/publications/ontology_development/ontology101.pdf.

[116] H. Osone, J.-L. Lu and Y. Ochiai, BunCho: ai supported story co-creation via unsupervised multitask learning to increase writers' creativity in japanese, in: *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021, pp. 1–10.

[117] F. Paas, A. Renkl and J. Sweller, Cognitive load theory and instructional design: Recent developments, *Educational psychologist* **38**(1) (2003), 1–4.

[118] M.J. Page, J.E. McKenzie, P.M. Bossuyt, I. Boutron, T.C. Hoffmann, C.D. Mulrow, L. Shamseer, J.M. Tetzlaff, E.A. Akl, S.E. Brennan et al., The PRISMA 2020 statement: an updated guideline for reporting systematic reviews, *bmj* **372** (2021).

[119] J.Z. Pan, S. Razniewski, J.-C. Kalo, S. Singhania, J. Chen, S. Dietze, H. Jabeen, J. Omeliyanenko, W. Zhang, M. Lissandrini et al., Large language models and knowledge graphs: Opportunities and challenges, *arXiv preprint arXiv:2308.06374* (2023).

[120] S. Peroni, A simplified agile methodology for ontology development, in: *International Experiences and Directions Workshop on OWL*, Springer, 2016, pp. 55–69.

[121] M. Poveda-Villalón, A. Gómez-Pérez and M.C. Suárez-Figueroa, Oops!(ontology pitfall scanner!): An on-line tool for ontology evaluation, *International Journal on Semantic Web and Information Systems (IJSWIS)* **10**(2) (2014), 7–34.

[122] M. Poveda-Villalón, A. Fernández-Izquierdo, M. Fernández-López and R. García-Castro, LOT: An industrial oriented ontology engineering framework, *Engineering Applications of Artificial Intelligence* **111** (2022), 104755.

[123] V. Presutti, E. Daga, A. Gangemi and E. Blomqvist, eXtreme design with content ontology design patterns, in: *Proc. Workshop on Ontology Patterns*, 2009, pp. 83–97.

[124] H.X. Qin, S. Jin, Z. Gao, M. Fan and P. Hui, CharacterMeet: Supporting Creative Writers' Entire Story Character Construction Processes Through Conversation with LLM-Powered Chatbot Avatars, in: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–19.

[125] Y. Rebboud, L. Tailhardat, P. Lisena and R. Troncy, Can LLMs Generate Competency Questions?, in: *European Semantic Web Conference*, Springer Nature Switzerland, Cham, 2024, pp. 71–80, Springer.

[126] Y. Ren, A. Parvizi, C. Mellish, J.Z. Pan, K. Van Deemter and R. Stevens, Towards competency question-driven ontology authoring, in: *European Semantic Web Conference*, Springer, 2014, pp. 752–767.

[127] M. Reza, N.M. Laundry, I. Musabirov, P. Dushniku, Z.Y.M. Yu, K. Mittal, T. Grossman, M. Liut, A. Kuzminykh and J.J. Williams, ABScribe: Rapid Exploration & Organization of Multiple Writing Variations in Human-AI Co-Writing Tasks using Large Language Models, in: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–18.

[128] G.W. Ryan, H.R. Bernard et al., Data management and analysis methods, *Handbook of qualitative research* **2**(1) (2000), 769–802.
[129] M.J. Saeedizade and E. Blomqvist, Navigating ontology development with large language models, in: *European semantic web conference*, Springer, 2024, pp. 143–161.
[130] M. Sarkar and M.H. Brown, Graphical fisheye views, *Communications of the ACM* **37**(12) (1994), 73–83.
[131] F. Sclano, P. Velardi et al., TermExtractor: A Web application to learn the common terminology of interest groups and research communities, in: *Proceedings of the 9th Conference on Terminology and Artificial Intelligence*, Citeseer, 2007, pp. 1–10.
[132] J.F. Sequeda, W.J. Briggs, D.P. Miranker and W.P. Heideman, A pay-as-you-go methodology to design and build enterprise knowledge graphs from relational databases, in: *International semantic web conference*, Springer, 2019, pp. 526–545.
[133] N. Shadbolt and A.M. Burton, The empirical study of knowledge elicitation techniques, *ACM SIGART Bulletin* (1989), 15–18.
[134] M. Shaw, The role of design spaces, *IEEE software* **29**(1) (2011), 46–50.
[135] Y. Shi, T. Gao, X. Jiao and N. Cao, Understanding design collaboration between designers and artificial intelligence: a systematic literature review, *Proceedings of the ACM on Human-Computer Interaction* **7**(CSCW2) (2023), 1–35.
[136] B. SHNEIDERMAN, The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations, *VL96* (1996), 336–343.
[137] A.P. Siddaway, A.M. Wood and L.V. Hedges, How to do a systematic review: a best practice guide for conducting and reporting narrative reviews, meta-analyses, and meta-syntheses, *Annual review of psychology* **70**(1) (2019), 747–770.
[138] E. Simperl and M. Luczak-Rösch, Collaborative ontology engineering: a survey, *The Knowledge Engineering Review* **29**(1) (2014), 101–131.
[139] N. Singh, G. Bernal, D. Savchenko and E.L. Glassman, Where to hide a stolen elephant: Leaps in creative writing with multimodal machine intelligence, *ACM Transactions on Computer-Human Interaction* **30**(5) (2023), 1–57.
[140] K. Son, D. Choi, T.S. Kim, Y.-H. Kim and J. Kim, Genquery: Supporting expressive visual search with generative models, in: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–19.
[141] S. Staab, R. Studer, H.-P. Schnurr and Y. Sure, Knowledge processes and ontologies, *IEEE Intelligent systems* **16**(1) (2001), 26–34.
[142] B.J. Stucky, J.P. Balhoff, N. Barve, V. Barve, L. Brenskelle, M.H. Brush, G.A. Dahlem, J.D. Gilbert, A.Y. Kawahara, O. Keller et al., Developing a vocabulary and ontology for modeling insect natural history data: example data, use cases, and competency questions, *Biodiversity data journal* **7** (2019), e33303.
[143] M.C. Suárez-Figueroa, A. Gómez-Pérez and M. Fernandez-Lopez, The NeOn Methodology framework: A scenario-based methodology for ontology development, *Applied ontology* **10**(2) (2015), 107–145.
[144] S. Suh, B. Min, S. Palani and H. Xia, Sensecape: Enabling multilevel exploration and sensemaking with large language models, in: *Proceedings of the 36th annual ACM symposium on user interface software and technology*, 2023, pp. 1–18.
[145] S. Suh, M. Chen, B. Min, T.J.-J. Li and H. Xia, Luminate: Structured generation and exploration of design space with large language models for human-ai co-creation, in: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–26.
[146] N. Sultanum and A. Srinivasan, Datatales: Investigating the use of large language models for authoring data-driven articles, in: *2023 IEEE Visualization and Visual Analytics (VIS)*, IEEE, 2023, pp. 231–235.
[147] Y. Sun, Z. Li, K. Fang, C.H. Lee and A. Asadipour, Language as reality: a co-creative storytelling game experience in 1001 nights using generative AI, in: *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Vol. 19, 2023, pp. 425–434.
[148] B. Swanson, K. Mathewson, B. Pietrzak, S. Chen and M. Dinalescu, Story centaur: Large language model few shot learning as a creative writing tool, in: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, 2021, pp. 244–256.
[149] J. Sweller, Cognitive load during problem solving: Effects on learning, *Cognitive science* **12**(2) (1988), 257–285.
[150] A. Taheri, M. Izadi, G. Shriram, N. Rostamzadeh and S. Kane, Breaking barriers to creative expression: Co-designing and implementing an accessible text-to-image interface, *arXiv preprint arXiv:2309.02402* (2023).
[151] D.R. Thomas, A general inductive approach for analyzing qualitative evaluation data, *American journal of evaluation* **27**(2) (2006), 237–246.
[152] J. Tidwell, *Designing interfaces: Patterns for effective interaction design*, " O'Reilly Media, Inc.", 2010.
[153] B. Tilekbay, S. Yang, M.A. Lewkowicz, A. Suryapranata and J. Kim, ExpressEdit: Video Editing with Natural Language and Sketching, in: *Proceedings of the 29th International Conference on Intelligent User Interfaces*, 2024, pp. 515–536.
[154] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser and I. Polosukhin, Attention is all you need, *Advances in neural information processing systems* **30** (2017).
[155] J.C. Vidal, T. Rabelo, M. Lama and R. Amorim, Ontology-based approach for the validation and conformance testing of xAPI events, *Knowledge-Based Systems* **155** (2018), 22–34.
[156] D. Vrandečić, L. Pintscher and M. Krötzsch, Wikidata: The making of, in: *Companion Proceedings of the ACM Web Conference 2023*, ACM, Austin, 2023, pp. 615–624.
[157] Q. Wan, X. Feng, Y. Bei, Z. Gao and Z. Lu, Metamorpheus: Interactive, Affective, and Creative Dream Narration Through Metaphorical Visual Storytelling, in: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–16.
[158] B. Wang, Y. Li, Z. Lv, H. Xia, Y. Xu and R. Sodhi, LAVE: LLM-powered agent assistance and language augmentation for video editing, in: *Proceedings of the 29th International Conference on Intelligent User Interfaces*, 2024, pp. 699–714.
[159] S. Wang, S. Petridis, T. Kwon, X. Ma and L.B. Chilton, PopBlends: Strategies for conceptual blending with large language models, in: *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, 2023, pp. 1–19.
[160] S. Wang, Z. Ning, A. Truong, M. Dontcheva, D. Li and L.B. Chilton, PodReels: Human-AI Co-Creation of Video Podcast Teasers, in: *Proceedings of the 2024 ACM Designing Interactive Systems Conference*, 2024, pp. 958–974.

[161] S. Wang, S. Menon, T. Long, K. Henderson, D. Li, K. Crowston, M. Hansen, J.V. Nickerson and L.B. Chilton, ReelFramer: Human-AI co-creation for news-to-video translation, in: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–20.

[162] Z. Wang, Y. Huang, D. Song, L. Ma and T. Zhang, Promptcharm: Text-to-image generation through multi-modal prompting and refinement, in: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–21.

[163] C. Ware and S. Osborne, Exploration and virtual camera control in virtual three dimensional environments, in: *Proceedings of the 1990 symposium on Interactive 3D graphics*, 1990, pp. 175–183.

[164] F. Weidt and R. Silva, Systematic literature review in computer science-a practical guide, *Relatórios Técnicos Do DCC/UFJF* **1**(8) (2016), 1–7.

[165] B. Westerlund, Design space conceptual tool–grasping the design process, *Nordes* (2005).

[166] M. Wiberg and E. Stolterman, What makes a prototype novel? A knowledge contribution concern for interaction design research, in: *Proceedings of the 8th Nordic conference on human-computer interaction: fun, fast, foundational*, 2014, pp. 531–540.

[167] R.H. Willemsen, I.C. de Vink, E.H. Kroesbergen and A.W. Lazonder, The role of creative thinking in children's scientific reasoning, *Thinking Skills and Creativity* **49** (2023), 101375.

[168] C. Williamson and B. Shneiderman, The Dynamic HomeFinder: Evaluating dynamic queries in a real-estate information exploration system, in: *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, 1992, pp. 338–346.

[169] D. Wiśniewski, J. Potoniec, A. Ławrynowicz and C.M. Keet, Analysis of ontology competency questions and their formalizations in SPARQL-OWL, *Journal of Web Semantics* **59** (2019), 100534.

[170] J.O. Wobbrock and J.A. Kientz, Research contributions in human-computer interaction, *interactions* **23**(3) (2016), 38–44.

[171] C. Wohlin, Guidelines for snowballing in systematic literature studies and a replication in software engineering, in: *Proceedings of the 18th international conference on evaluation and assessment in software engineering*, 2014, pp. 1–10.

[172] L. Xie, C. Zheng, H. Xia, H. Qu and C. Zhu-Tian, Waitgpt: Monitoring and steering conversational llm agent in data analysis with on-the-fly code visualization, in: *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, 2024, pp. 1–14.

[173] H. Yakura and M. Goto, IteraTTA: An interface for exploring both text prompts and audio priors in generating music with text-to-audio models, *arXiv preprint arXiv:2307.13005* (2023).

[174] Z. Yan, C. Yang, Q. Liang and X. Chen, XCreation: A graph-based crossmodal generative creativity support tool, in: *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, 2023, pp. 1–15.

[175] D. Yang, E. Kleinman, G.M. Troiano, E. Tochilnikova and C. Harteveld, Snake Story: Exploring Game Mechanics for Mixed-initiative Co-creative Storytelling Games, in: *Proceedings of the 19th International Conference on the Foundations of Digital Games*, 2024, pp. 1–11.

[176] J. Yang, K. Mo, Y.-K. Lai, L.J. Guibas and L. Gao, DSG-Net: Learning disentangled structure and geometry for 3D shape generation, *ACM Transactions on Graphics (TOG)* **42**(1) (2022), 1–17.

[177] J.S. Yi, Y. ah Kang, J. Stasko and J.A. Jacko, Toward a deeper understanding of the role of interaction in information visualization, *IEEE transactions on visualization and computer graphics* **13**(6) (2007), 1224–1231.

[178] R.J. Youmans and T. Arciszewski, Design fixation: Classifications and modern methods of prevention, *AI EDAM* **28**(2) (2014), 129–137.

[179] A. Yuan, A. Coenen, E. Reif and D. Ippolito, Wordcraft: story writing with large language models, in: *Proceedings of the 27th International Conference on Intelligent User Interfaces*, 2022, pp. 841–852.

[180] B. Zhang, V.A. Carriero, K. Schreiberhuber, S. Tsaneva, L.S. González, J. Kim and J. de Berardinis, OntoChat: a framework for conversational ontology engineering using language models, in: *European Semantic Web Conference*, Springer, 2024, pp. 102–121.

[181] Z. Zhang, J. Gao, R.S. Dhaliwal and T.J.-J. Li, Visar: A human-ai argumentative writing assistant with visual programming and rapid draft prototyping, in: *Proceedings of the 36th annual ACM symposium on user interface software and technology*, 2023, pp. 1–30.

[182] Y. Zhao, Leveraging large language models for ontology requirements engineering, in: *European Semantic Web Conference*, Springer, 2025, pp. 254–264.

[183] Y. Zhao, A. Meroño Peñuela and E. Simperl, OntoScope: Using a Divergent-Convergent Interaction Framework to Support LLM-based Ontology Scoping, in: *Proceedings of the 31st International Conference on Intelligent User Interfaces*, 2026, pp. 67–84.

[184] Y. Zhao, A.M. Peñuela and E. Simperl, User experience in dataset search, in: *International Conference on Computer-Human Interaction Research and Applications*, Springer, 2024, pp. 113–130.

[185] Y. Zhao, B. Zhang, X. Hu, S. Ouyang, J. Kim, N. Jain, J. de Berardinis, A. Meroño-Peñuela and E. Simperl, Improving Ontology Requirements Engineering with OntoChat and Participatory Prompting, in: *Proceedings of the AAAI Symposium Series*, Vol. 4, 2024, pp. 253–257.

[186] Y. Zhao, Z. Li, Y. Wang, X. Cai, X. Zhou, Y. Yan, K. Jin, S. Ding, Y. Shao, J. Cao et al., DreamDirector: Designing a Generative AI System to Aid Therapists in Treating Clients' Nightmares, in: *Proceedings of the 30th International Conference on Intelligent User Interfaces*, 2025, pp. 553–578.

[187] A. Zhu, L. Martin, A. Head and C. Callison-Burch, CALYPSO: LLMs as Dungeon Master's Assistants, in: *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Vol. 19, 2023, pp. 380–390.