Uta Priss
Simon Polovina
Richard Hill (Eds.)

LNAI 4604

Conceptual Structures: Knowledge Architectures for Smart Applications

15th International Conference on Conceptual Structures, ICCS 2007
Sheffield, UK, July 2007, Proceedings

Springer

Uta Priss Simon Polovina
Richard Hill (Eds.)

Conceptual Structures: Knowledge Architectures for Smart Applications

15th International Conference
on Conceptual Structures, ICCS 2007
Sheffield, UK, July 22-27, 2007
Proceedings

Springer

Series Editors

Jaime G. Carbonell, Carnegie Mellon University, Pittsburgh, PA, USA
Jörg Siekmann, University of Saarland, Saarbrücken, Germany

Volume Editors

Uta Priss
Napier University, School of Computing
10 Colinton Road, Edinburgh, EH10 5DT, UK
E-mail: u.priss@napier.ac.uk

Simon Polovina
Sheffield Hallam University, City Campus, Harmer Building
Howard Street, Sheffield, S1 1WB, UK
E-mail: s.polovina@shu.ac.uk

Richard Hill
Sheffield Hallam University, City Campus, Stoddart Building
Howard Street, Sheffield, S1 1WB, UK
E-mail: r.hill@shu.ac.uk

Library of Congress Control Number: 2007930463

CR Subject Classification (1998): I.2, G.2.2, F.4.1, F.2.1, H.4

LNCS Sublibrary: SL 7 – Artificial Intelligence

ISSN 0302-9743
ISBN-10 3-540-73680-8 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-73680-6 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12092222 06/3180 5 4 3 2 1 0

Lecture Notes in Artificial Intelligence 4604

Edited by J. G. Carbonell and J. Siekmann

Subseries of Lecture Notes in Computer Science

Preface

This volume contains the proceedings of the 15th International Conference on Conceptual Structures (ICCS 2007), which is an annual event that, for the first time, was hosted in the UK. Conceptual structures focus on the representation and analysis of concepts, events, actions and objects with applications in research, software engineering, manufacturing and business. The conference brings together researchers in computer science, information technology, artificial intelligence, philosophy and a variety of applied disciplines to explore novel ways that information technologies can be leveraged to assist human reasoning and interaction for tangible business or social benefits. Conceptual structures can be used to augment human intelligence by facilitating knowledge integration, decision making, the creation of intelligent software systems and the exploration of implicit structures.

The theme for this year's conference was "Conceptual Structures: Knowledge Architectures for Smart Applications." Knowledge architectures give rise to smart applications that allow enterprises to share meaning across their interconnected computing resources and to realize transactions that would otherwise remain as lost business opportunities. Conceptual structures and smart applications integrate the creativity of individuals and organizations with the productivity of computers for a meaningful digital future. A focus of ICCS 2007 was on papers that apply conceptual structures in business and technological settings. Other submitted papers covered research in conceptual structures, which is supported by mathematical and computational theory, including formal concept analysis, algorithm design and graph theory, and a variety of software tools.

The conference had a rigorous refereeing process. All papers were reviewed by one Editorial Board member and two Program Committee members. About 50% of the submitted papers were accepted as full papers to be presented at the conference. A few additional papers were accepted as short or position papers. In addition, four invited papers and one introductory paper by Simon Polovina are included in this proceedings volume.

We wish to express our thanks to all the authors of submitted papers, to the members of the Editorial Board and Program Committee, to the Workshop and Tutorial Chair, B. Akhgar, the Industry Chair, J. Schiffel, and the Sponsorship Chair, D. Corbett, and to our sponsors!

July 2007

Uta Priss
Simon Polovina
Richard Hill

Organization

The International Conference on Conceptual Structures (ICCS) is the annual conference and principal research forum in the theory and practice of conceptual structures. The conference has been held since 1993 in various locations: Universitè Laval (Quebec City, 1993), University of Maryland (1994), University of California (Santa Cruz, 1995), Sydney (1996), University of Washington (Seattle, 1997), Montpellier (1998), Virginia Tech (Blacksburg, 1999), Technische Universität Darmstadt (2000), Stanford University (2001), Borovets (Bulgaria, 2002), Technische Universität Dresden (2003), University of Alabama (Huntsville, 2004), Universität Kassel (2005) and Aalborg University (Denmark, 2006).

General Chair

Simon Polovina — Sheffield Hallam University, UK

Program Chairs

Uta Priss — Napier University, Edinburgh, UK
Richard Hill — Sheffield Hallam University, UK

Other ICCS Chairs

Jeffrey Schiffel (Industry Chair)
Babak Akhgar (Workshop Chair)
Daniel Corbett (Sponsorship Chair)

Editorial Board

Program Committee

Radim Bělohlávek (Czech Republic)
Tru Cao (Vietnam)
Pavlin Dobrev (Bulgaria)
David Genest (France)
Udo Hebisch (Germany)
Wolfgang Hesse (Germany)
Christian Jacquelinet (France)
Pavel Kocura (UK)
Robert Kremer (Canada)
Leonhard Kwuida (Switzerland)
Robert Levinson (USA)
Carsten Lutz (Germany)
Claudio Masolo (Italy)
Jørgen Fischer Nilsson (Denmark)
Ulrik Petersen (Denmark)
Gary Richmond (USA)
Sebastian Rudolph (Germany)
Janos Sarbo (The Netherlands)
GQ Zhang (USA)
Anne Berry (France)
Dan Corbett (Australia)
Peter Eklund (Australia)
Ollivier Haemmerle (France)
Joachim Hereth Correia (Germany)
Andreas Hotho (Germany)
Adil Kabbaj (Morocco)
Yannis Kalfoglou (UK)
Markus Krötzsch (Germany)
Michel Leclère (France)
Michel Liquière (France)
Philippe Martin (Australia)
Engelbert Mephu Nguifo (France)
Sergei Obiedkov (South Africa)
Anne-Marie Rassinoux (Switzerland)
Olivier Ridoux (France)
Éric Salvat (France)
William Tepfenhart (USA)

Further Reviewers

Simone Braun (Germany)
Rainer Osswald (Germany)
Yimin Wang (Germany)
L. John Old (UK)
Quan Thanh Tho (Vietnam)

Sponsoring Institutions

Cultural, Communication and Computing Research Institute (C3RI), Sheffield Hallam University, UK
EU IST MOSIACA Project
Institute of Engineering and Technology, South Yorkshire Network, UK
Natural Language Processing Research Group, Department of Computer Science, University of Sheffield, UK
British Computer Society, South Yorkshire Branch

Lecture Notes in Artificial Intelligence (LNAI)

Vol. 4612: I. Miguel, W. Ruml (Eds.), Abstraction, Reformulation, and Approximation. XI, 418 pages. 2007.

Vol. 4604: U. Priss, S. Polovina, R. Hill (Eds.), Conceptual Structures: Knowledge Architectures for Smart Applications. XII, 514 pages. 2007.

Vol. 4597: P. Perner (Ed.), Advances in Data Mining. XI, 353 pages. 2007.

Vol. 4594: R. Bellazzi, A. Abu-Hanna, J. Hunter (Eds.), Artificial Intelligence in Medicine. XVI, 509 pages. 2007.

Vol. 4585: M. Kryszkiewicz, J.F. Peters, H. Rybinski, A. Skowron (Eds.), Rough Sets and Intelligent Systems Paradigms. XIX, 836 pages. 2007.

Vol. 4578: F. Masulli, S. Mitra, G. Pasi (Eds.), Applications of Fuzzy Sets Theory. XVIII, 693 pages. 2007.

Vol. 4573: M. Kauers, M. Kerber, R. Miner, W. Windsteiger (Eds.), Towards Mechanized Mathematical Assistants. XIII, 407 pages. 2007.

Vol. 4571: P. Perner (Ed.), Machine Learning and Data Mining in Pattern Recognition. XIV, 913 pages. 2007.

Vol. 4570: H.G. Okuno, M. Ali (Eds.), New Trends in Applied Artificial Intelligence. XXI, 1194 pages. 2007.

Vol. 4565: D.D. Schmorrow, L.M. Reeves (Eds.), Foundations of Augmented Cognition. XIX, 450 pages. 2007.

Vol. 4562: D. Harris (Ed.), Engineering Psychology and Cognitive Ergonomics. XXIII, 879 pages. 2007.

Vol. 4548: N. Olivetti (Ed.), Automated Reasoning with Analytic Tableaux and Related Methods. X, 245 pages. 2007.

Vol. 4539: N.H. Bshouty, C. Gentile (Eds.), Learning Theory. XII, 634 pages. 2007.

Vol. 4529: P. Melin, O. Castillo, L.T. Aguilar, J. Kacprzyk, W. Pedrycz (Eds.), Foundations of Fuzzy Logic and Soft Computing. XIX, 830 pages. 2007.

Vol. 4511: C. Conati, K. McCoy, G. Paliouras (Eds.), User Modeling 2007. XVI, 487 pages. 2007.

Vol. 4509: Z. Kobti, D. Wu (Eds.), Advances in Artificial Intelligence. XII, 552 pages. 2007.

Vol. 4496: N.T. Nguyen, A. Grzech, R.J. Howlett, L.C. Jain (Eds.), Agent and Multi-Agent Systems: Technologies and Applications. XXI, 1046 pages. 2007.

Vol. 4483: C. Baral, G. Brewka, J. Schlipf (Eds.), Logic Programming and Nonmonotonic Reasoning. IX, 327 pages. 2007.

Vol. 4482: A. An, J. Stefanowski, S. Ramanna, C.J. Butz, W. Pedrycz, G. Wang (Eds.), Rough Sets, Fuzzy Sets, Data Mining and Granular Computing. XIV, 585 pages. 2007.

Vol. 4481: J. Yao, P. Lingras, W.-Z. Wu, M. Szczuka, N.J. Cercone, D. Ślęzak (Eds.), Rough Sets and Knowledge Technology. XIV, 576 pages. 2007.

Vol. 4476: V. Gorodetsky, C. Zhang, V.A. Skormin, L. Cao (Eds.), Autonomous Intelligent Systems: Multi-Agents and Data Mining. XIII, 323 pages. 2007.

Vol. 4452: M. Fasli, O. Shehory (Eds.), Agent-Mediated Electronic Commerce. VIII, 249 pages. 2007.

Vol. 4451: T.S. Huang, A. Nijholt, M. Pantic, A. Pentland (Eds.), Artifical Intelligence for Human Computing. XVI, 359 pages. 2007.

Vol. 4438: L. Maicher, A. Sigel, L.M. Garshol (Eds.), Leveraging the Semantics of Topic Maps. X, 257 pages. 2007.

Vol. 4429: R. Lu, J.H. Siekmann, C. Ullrich (Eds.), Cognitive Systems. X, 161 pages. 2007.

Vol. 4426: Z.-H. Zhou, H. Li, Q. Yang (Eds.), Advances in Knowledge Discovery and Data Mining. XXV, 1161 pages. 2007.

Vol. 4411: R.H. Bordini, M. Dastani, J. Dix, A.E.F. Seghrouchni (Eds.), Programming Multi-Agent Systems. XIV, 249 pages. 2007.

Vol. 4410: A. Branco (Ed.), Anaphora: Analysis, Algorithms and Applications. X, 191 pages. 2007.

Vol. 4399: T. Kovacs, X. Llorà, K. Takadama, P.L. Lanzi, W. Stolzmann, S.W. Wilson (Eds.), Learning Classifier Systems. XII, 345 pages. 2007.

Vol. 4390: S.O. Kuznetsov, S. Schmidt (Eds.), Formal Concept Analysis. X, 329 pages. 2007.

Vol. 4389: D. Weyns, H.V.D. Parunak, F. Michel (Eds.), Environments for Multi-Agent Systems III. X, 273 pages. 2007.

Vol. 4384: T. Washio, K. Satoh, H. Takeda, A. Inokuchi (Eds.), New Frontiers in Artificial Intelligence. IX, 401 pages. 2007.

Vol. 4371: K. Inoue, K. Satoh, F. Toni (Eds.), Computational Logic in Multi-Agent Systems. X, 315 pages. 2007.

Vol. 4369: M. Umeda, A. Wolf, O. Bartenstein, U. Geske, D. Seipel, O. Takata (Eds.), Declarative Programming for Knowledge Management. X, 229 pages. 2006.

Vol. 4342: H. de Swart, E. Orłowska, G. Schmidt, M. Roubens (Eds.), Theory and Applications of Relational Structures as Knowledge Instruments II. X, 373 pages. 2006.

Vol. 4335: S.A. Brueckner, S. Hassas, M. Jelasity, D. Yamins (Eds.), Engineering Self-Organising Systems. XII, 212 pages. 2007.

Vol. 4334: B. Beckert, R. Hähnle, P.H. Schmitt (Eds.), Verification of Object-Oriented Software. XXIX, 658 pages. 2007.

Vol. 4333: U. Reimer, D. Karagiannis (Eds.), Practical Aspects of Knowledge Management. XII, 338 pages. 2006.

Vol. 4327: M. Baldoni, U. Endriss (Eds.), Declarative Agent Languages and Technologies IV. VIII, 257 pages. 2006.

Vol. 4314: C. Freksa, M. Kohlhase, K. Schill (Eds.), KI 2006: Advances in Artificial Intelligence. XII, 458 pages. 2007.

Vol. 4304: A. Sattar, B.-H. Kang (Eds.), AI 2006: Advances in Artificial Intelligence. XXVII, 1303 pages. 2006.

Vol. 4303: A. Hoffmann, B.-H. Kang, D. Richards, S. Tsumoto (Eds.), Advances in Knowledge Acquisition and Management. XI, 259 pages. 2006.

Vol. 4293: A. Gelbukh, C.A. Reyes-Garcia (Eds.), MICAI 2006: Advances in Artificial Intelligence. XXVIII, 1232 pages. 2006.

Vol. 4289: M. Ackermann, B. Berendt, M. Grobelnik, A. Hotho, D. Mladenič, G. Semeraro, M. Spiliopoulou, G. Stumme, V. Svátek, M. van Someren (Eds.), Semantics, Web and Mining. X, 197 pages. 2006.

Vol. 4285: Y. Matsumoto, R.W. Sproat, K.-F. Wong, M. Zhang (Eds.), Computer Processing of Oriental Languages. XVII, 544 pages. 2006.

Vol. 4274: Q. Huo, B. Ma, E.-S. Chng, H. Li (Eds.), Chinese Spoken Language Processing. XXIV, 805 pages. 2006.

Vol. 4265: L. Todorovski, N. Lavrač, K.P. Jantke (Eds.), Discovery Science. XIV, 384 pages. 2006.

Vol. 4264: J.L. Balcázar, P.M. Long, F. Stephan (Eds.), Algorithmic Learning Theory. XIII, 393 pages. 2006.

Vol. 4259: S. Greco, Y. Hata, S. Hirano, M. Inuiguchi, S. Miyamoto, H.S. Nguyen, R. Słowiński (Eds.), Rough Sets and Current Trends in Computing. XXII, 951 pages. 2006.

Vol. 4253: B. Gabrys, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part III. XXXII, 1301 pages. 2006.

Vol. 4252: B. Gabrys, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part II. XXXIII, 1335 pages. 2006.

Vol. 4251: B. Gabrys, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part I. LXVI, 1297 pages. 2006.

Vol. 4248: S. Staab, V. Svátek (Eds.), Managing Knowledge in a World of Networks. XIV, 400 pages. 2006.

Vol. 4246: M. Hermann, A. Voronkov (Eds.), Logic for Programming, Artificial Intelligence, and Reasoning. XIII, 588 pages. 2006.

Vol. 4223: L. Wang, L. Jiao, G. Shi, X. Li, J. Liu (Eds.), Fuzzy Systems and Knowledge Discovery. XXVIII, 1335 pages. 2006.

Vol. 4213: J. Fürnkranz, T. Scheffer, M. Spiliopoulou (Eds.), Knowledge Discovery in Databases: PKDD 2006. XXII, 660 pages. 2006.

Vol. 4212: J. Fürnkranz, T. Scheffer, M. Spiliopoulou (Eds.), Machine Learning: ECML 2006. XXIII, 851 pages. 2006.

Vol. 4211: P. Vogt, Y. Sugita, E. Tuci, C.L. Nehaniv (Eds.), Symbol Grounding and Beyond. VIII, 237 pages. 2006.

Vol. 4203: F. Esposito, Z.W. Raś, D. Malerba, G. Semeraro (Eds.), Foundations of Intelligent Systems. XVIII, 767 pages. 2006.

Vol. 4201: Y. Sakakibara, S. Kobayashi, K. Sato, T. Nishino, E. Tomita (Eds.), Grammatical Inference: Algorithms and Applications. XII, 359 pages. 2006.

Vol. 4200: I.F.C. Smith (Ed.), Intelligent Computing in Engineering and Architecture. XIII, 692 pages. 2006.

Vol. 4198: O. Nasraoui, O. Zaïane, M. Spiliopoulou, B. Mobasher, B. Masand, P.S. Yu (Eds.), Advances in Web Mining and Web Usage Analysis. IX, 177 pages. 2006.

Vol. 4196: K. Fischer, I.J. Timm, E. André, N. Zhong (Eds.), Multiagent System Technologies. X, 185 pages. 2006.

Vol. 4188: P. Sojka, I. Kopeček, K. Pala (Eds.), Text, Speech and Dialogue. XV, 721 pages. 2006.

Vol. 4183: J. Euzenat, J. Domingue (Eds.), Artificial Intelligence: Methodology, Systems, and Applications. XIII, 291 pages. 2006.

Vol. 4180: M. Kohlhase, OMDoc – An Open Markup Format for Mathematical Documents [version 1.2]. XIX, 428 pages. 2006.

Vol. 4177: R. Marín, E. Onaindía, A. Bugarín, J. Santos (Eds.), Current Topics in Artificial Intelligence. XV, 482 pages. 2006.

Vol. 4160: M. Fisher, W. van der Hoek, B. Konev, A. Lisitsa (Eds.), Logics in Artificial Intelligence. XII, 516 pages. 2006.

Vol. 4155: O. Stock, M. Schaerf (Eds.), Reasoning, Action and Interaction in AI Theories and Systems. XVIII, 343 pages. 2006.

Vol. 4149: M. Klusch, M. Rovatsos, T.R. Payne (Eds.), Cooperative Information Agents X. XII, 477 pages. 2006.

Vol. 4140: J.S. Sichman, H. Coelho, S.O. Rezende (Eds.), Advances in Artificial Intelligence - IBERAMIA-SBIA 2006. XXIII, 635 pages. 2006.

Vol. 4139: T. Salakoski, F. Ginter, S. Pyysalo, T. Pahikkala (Eds.), Advances in Natural Language Processing. XVI, 771 pages. 2006.

Vol. 4133: J. Gratch, M. Young, R. Aylett, D. Ballin, P. Olivier (Eds.), Intelligent Virtual Agents. XIV, 472 pages. 2006.

Vol. 4130: U. Furbach, N. Shankar (Eds.), Automated Reasoning. XV, 680 pages. 2006.

Vol. 4120: J. Calmet, T. Ida, D. Wang (Eds.), Artificial Intelligence and Symbolic Computation. XIII, 269 pages. 2006.

Vol. 4118: Z. Despotovic, S. Joseph, C. Sartori (Eds.), Agents and Peer-to-Peer Computing. XIV, 173 pages. 2006.

¥689.00元

Table of Contents

Invited Papers

Conceptual Graphs

Formal Concept Analysis

Conceptual Structures

Short Papers

An Introduction to Conceptual Graphs

Simon Polovina

Culture, Communication and Computing Research Institute (C3RI)
Faculty of Arts, Computing, Engineering & Sciences
Sheffield Hallam University, UK S1 1WB
s.polovina@shu.ac.uk

Abstract. This paper provides a lucid introduction to Conceptual Graphs (CG), a powerful knowledge representation and inference environment that exhibits the familiar object-oriented features of contemporary enterprise and web applications. An illustrative business case study is used to convey how CG adds value to data, including inference for new knowledge. It enables newcomers to conceptual structures to engage with this exciting field and to realise "Conceptual Structures: Knowledge Architectures for Smart Applications", the theme of the 15th Annual International Conference on Conceptual Structures (ICCS 2007, www.iccs2007.info).

1 Introduction

Conceptual Graphs (CG, www.conceptualgraphs.org) provide a powerful knowledge representation and inference environment, whilst exhibiting the familiar object-oriented and database features of contemporary enterprise and web applications. CG capture nuances in natural language whilst being able to be implemented in computer software. CG were devised by Sowa from philosophical, psychological, linguistic, and artificial intelligence foundations in a principled way [8, 9]. Hence CG are particularly attractive as they are built upon such a strong theoretical and wide-ranging base.

There is an active CG community, evidenced by the annual International Conferences on Conceptual Structures (ICCS, www.iccs.info), now in its 15th year (ICCS 2007, www.iccs2007.info), not to mention the annual CG workshops beforehand (www.conceptualstructures.org/confs.htm). There is also the CG discussion list (cg@conceptualgraphs.org). Its participants happily support newcomers to CG e.g. in answering queries; www.conceptualgraphs.org provides information on how to join this list, as well as a comprehensive catalogue of software CG tools.

CG is core to the ISO Common Logic standard (http://cl.tamu.edu/)[1]. The CG community has furthermore grown to embody a wider notion of Conceptual Structures (CS, www.conceptualstructures.org, and the title of Sowa's seminal 1984 text [9]). This is typified by the large scale and valued contributions that Formal Concepts Analysis (FCA, www.upriss.org.uk/fca) brings to ICCS each year [4].

[1] Assigned to WG2 (Metadata) under SC32 (Data Interchange) of ISO/IEC JTC1.

U. Priss, S. Polovina, and R. Hill (Eds.): ICCS 2007, LNAI 4604, pp. 1–14, 2007.

A strong case therefore exists for bringing an awareness of CG to an even wider community. Through an illustrative business case study, the following explication of CG[2] aims to achieve this objective so that many more researchers and industry professionals can realise the benefits of CG and, as the theme of ICCS 2004 highlighted, put them to work [10].

2 Concepts and Relations

CG are based upon the following general form:

This may be read as: "*The* relation *of a* Concept-1 *is a* Concept-2". The direction of the arrows assists the direction of the reading. If the arrows were pointing the other way, then the reading would be the same except that Concept-1 and Concept-2 would exchange places (i.e. "*The* relation *of* Concept-2 *is a* Concept-1").

As an alternative to the above 'display' form[3], the graphs may be written in the following 'linear' text-based form:

```
[Concept_1] -> (relation) -> [Concept_2].
```

The full stop '.' signals the end of a particular graph. Consider the following example:

```
[Funding_Request] -> (initiator) -> [Employee].
```

This example will form a part of an illustrative case study about requests for funding new business projects in the fictitious enterprise 'P-H Co.'. The example graph reads as "The initiator of a funding request is an employee". This may create readings that may sound long-winded or ungrammatical, but is a useful mnemonic aid in constructing and interpreting any graph. It is easier to state "An employee initiates a funding request".

Concepts can have *referents*, which refers to a particular instance, or individual, of that concept[4]. For example consider the concept:

```
[Employee: Simon].
```

This reads as "The employee known as Simon". The referent is a *conformity* to the *type label* in a concept. This example shows that Simon conforms to the type label 'Employee'.

---

[2] That also draws on an earlier introduction to CG in 1992 [7].

[3] The display form CG throughout this paper were produced using the *CharGer* software (http://charger.sourceforge.net/), one the many useful CG software tools that are catalogued at www.conceptualgraphs.org.

[4] There are other kinds of referents, such as plural ('sets' of) referents (which are rather like collections in object-orientated classes, and scalars ('measures') [8, 9]. In passing, as well as Concepts and Relations in CG there are Actors (which incidentally are not to be confused with UML Actors! [6]) Delugach is a proponent for the use of CG Actors [3].

A concept that appears without an individual referent has a *generic* referent. Such 'generic concepts' should be denoted as `[Type_Label: *]`. Writing `[Type_Label]` is merely a convenient shorthand.

Generic concepts may take up an individual referent. A unique identifier can be used to make a concept distinct. Thereby the generic concept `[Funding_Request]` might become `[Funding_Request: #1234]` with respect to `[Employee: Simon]`. This would yield:

```
[Funding_Request: #1234] -> (initiator) -> [Employee: Simon].
```

If there are two or more employees with the name Simon we would need to make them distinct from one another e.g. `[Employee: Simon#122014]`.

3 The Type Hierarchy

In CG, type labels belong to a type hierarchy. Thereby:

`Manager < Employee.` ("A manager is an employee")

This means Manager is a more specialised type of the type Employee i.e. Manager is a *subtype* of Employee. Alternatively, this can be stated as Employee is a *supertype* of Manager. (Subtypes and supertypes are analogous to subclasses and superclasses in object-orientation, thus a subtype inherits the characteristics of its supertypes.) Similarly, the remainder of the hierarchy may be:

```
Employee < Person.
Person < Animal.
Animal < Entity.
Entity < T.
Funding-Request < Request.
Request < Act.
Act < Event.
Act < T.
```

Sowa provides a conceptual catalog that includes a representative set of hierarchical concepts, as well as relations [9]. It also shows the context in which a type label should be used. For example an Act is an Event with an Animate agent:

```
[Act] -> (agent) -> [Animate].
```

The type denoted as 'T' means the universal supertype. It has no supertypes and is therefore the most general type.

A subtype can have more than one immediate supertype. For example, consider the concept `[Animal]`, which has a more detailed set of supertypes than indicated above. This concept has a type label Animal which may be defined as [9]:

```
Animal < Animate, Mobile_Entity, Physical_Object, ¬Machine.
Animate < Entity.
Mobile_Entity < Entity.
Physical_Object < Entity.
```

An animal therefore inherits the characteristics of being animate, a mobile-entity, and a physical object, but it is not a machine (¬ means 'not').

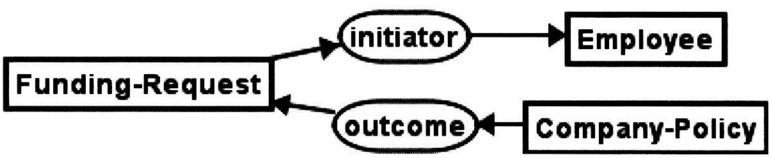
4 Projection

Projection in CG extends the notion of projection in SQL in database systems [1, 8, 9]. It is used in conjunction with specialised graphs. A CG becomes more specialised when one or more of the following happens:

a) more concepts, types and relations are added to it to narrow the scope further, e.g. an employee initiates a funding request that is the outcome of a company policy (as opposed to the earlier where it is only shown that an employee initiates a funding request) i.e.:

```
[Funding_Request] -
 (initiator) -> [Employee]
 (outcome) <- [Company_Policy].
```

This larger graph illustrates the use of hyphen '-', which allows the relations of a concept (being Funding-Request in this case) to be listed on subsequent lines. In the visual display form of CG this is not needed as the following equivalent CG shows:



b) it acquires non-generic referents, e.g. `[Employee]` becomes `[Employee: Simon]`, or
c) subtypes are substituted for (super)types e.g. replacing `[Employee]` with `[Manager]`

The following example illustrates a combination of these:

```
[Funding-request] -
 (initiator) -> [Manager: Susan]
 (outcome) <- [Company-Policy: #CP76321].
```

Therefore each specialised graph may have a more general graph from which it was derived. Likewise a general graph can have a number of specialised variants. Thus *projection* is where we take a graph and try to see if another graph is a generalisation of it. It there is such a fit we have determined that the graph is indeed a specialised variant, and that the other graph is indeed a generalisation of it. We can then state that the general graph 'projects' into the specialised one.