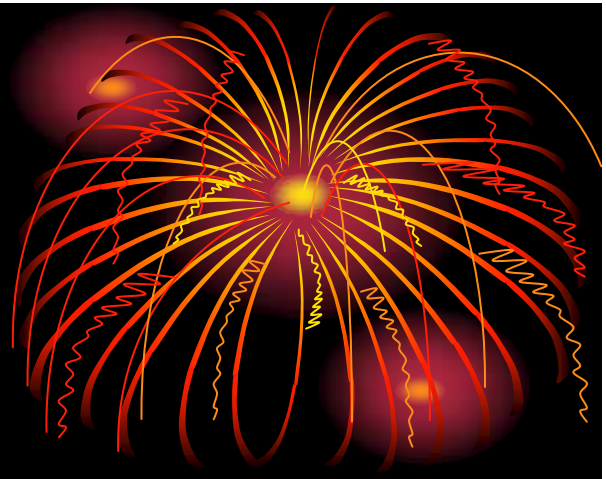
ER2004
Panel

Hot Research Topics: How Conceptual Modeling Can Contribute

Octavian Panel

- 8 panel members
- Only panel members speak
- Each comment < 2 minutes
- At most 2 comments before leaving
- Next audience member in queue joins panel

Objectives



- Raise hot-topic issues
- Understand how conceptual modeling applies
- Outline single-research-size projects
  - What's the problem/question?
  - Why is the problem/question interesting?
  - What's a resolution or direction to pursue?
- Generate lively discussion
- Stimulate research

Semantic Web

Process Ontologies for Services

- Problem (Dave Embley)
  - Can process ontologies be endowed with enough knowledge to flexibly guide service selection/invocation/completion?
  - E.g. medical appointment management ontology
- Why interesting
  - If so, we may be able to realize the quintessential vision of the semantic web
  - E.g. “Julie needs to see a doctor who accepts my insurance – as soon as possible.”
- Directions to pursue
  - Develop process ontologies and a prototype execution system that can match and execute free-form task specification
  - E.g. find out who Julie is, negotiate appointment with doctor, handle insurance issues, ...

Model-Driven Development

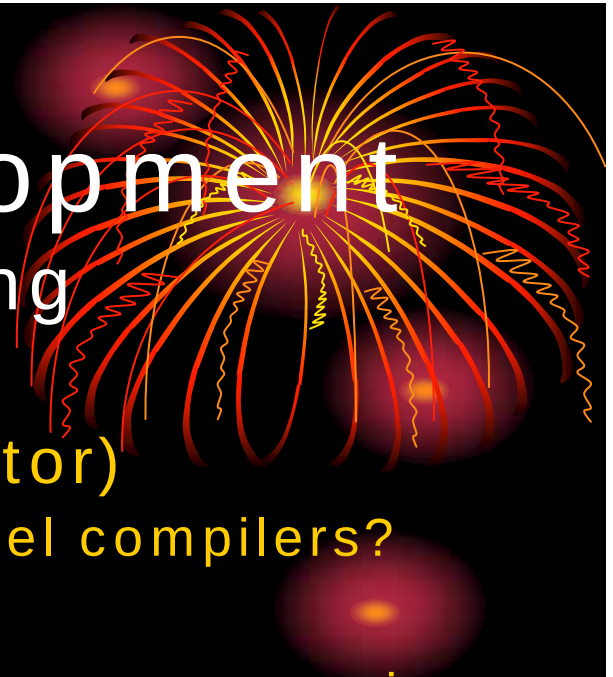
Conceptual Schemas ... And Nothing Else

- Problem (Antoni Olivé)
  - Fact: a lot of effort is needed to design, program, and test IS's
  - Conjecture: nothing prevents the automation of this effort
- Why interesting
  - There is an obvious (and big) economic interest in the solution of the problem
- Directions to pursue
  - Black box: information systems management systems
  - White box: code generation

Semantic Web Services
Project Title

- Problem (Erich Neuhold)
  - Need ontologies and models; include
- Why interesting
  - Web services will not work without solving this problem
- Directions to pursue
  - ...

Model-Driven Development
Extreme Conceptual Modeling



- Problem (Óscar Pastor)
  - Can we build true conceptual model compilers?
- Why interesting
  - MDA is increasingly important to many companies; but we still need better tools that have a precise ontological and formal background, yet with clear value for practitioners
- Directions to pursue
  - From requirements to software product, through model transformations, preserving functional equivalence, founded on a sound conceptual modeling basis

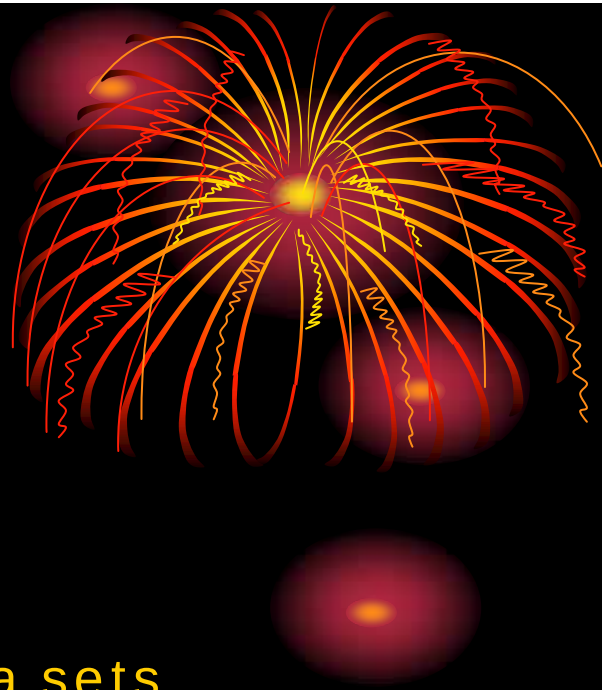
Context Awareness
Conceptual Modeling of Context

- Problem (Burkhard Freitag)
  - Context affects data and processes at various levels
- Why interesting
  - Given the web and enterprise needs, we must deploy applications that are sensitive to context across many dimensions
  - E.g. language, culture, hardware, network capabilities
- Directions to pursue
  - Learn how to model context (across dimensions of time, location, presentation mode, network configuration and capabilities, etc.)
  - Define something like “context middleware”

Web Services
Superimposed Systems

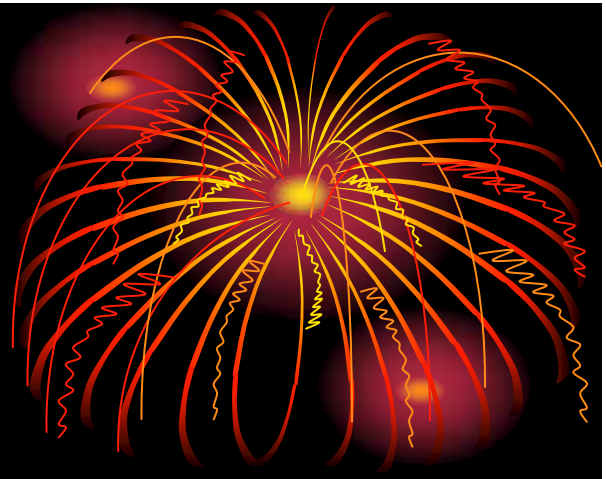
- The problem (Lois Delcambre)
  - How can we use other systems, services, information sources, ... as they are, where they are ...?
- Why interesting
  - Superimposed information (new information – with references to small bits of information) is a key element of web architecture
- Directions to pursue
  - What sort of conceptual models help us describe and construct composed and superimposed systems?

Brainstorming



- Possible hot topics
  - Security
  - Bioinformatics
  - Integration & querying across data sets
  - XML
  - Grid services/environment
    - Have been looking at this from perspective of distributed processing
    - How do we ensure quality, trust, privacy? We need to have models of these in order to make reasonable judgments
    - Tie models of services/data to these other models
  - Data quality – how to measure, ensure

Brainstorming

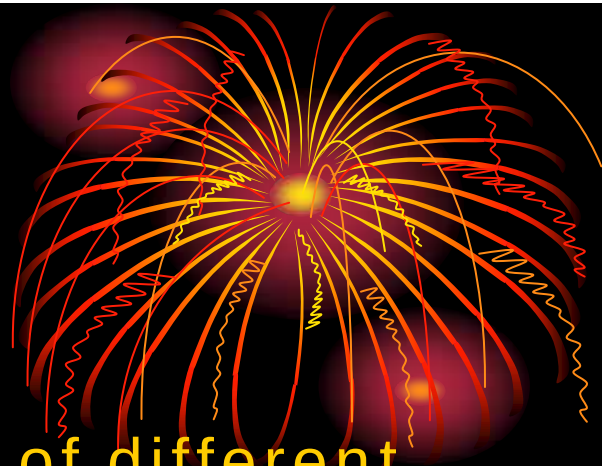


- Ubiquitous computing
  - Lots of corporations need this help
    - Marine Corps – logistics, just-in-time
  - Location-aware services is a really hard problem
  - Extending UML hasn't been successful
- At-home computing
  - End-users may become designers of software to an increasing degree. If so, they'll need CM tool support.
  - Don't just focus on business users
- Possible projects
  - …

Brainstorming

- How do you evaluate the quality of your work/research?
  - Devise objective, systematic evaluation
- Peer-to-peer semantic overlay networks
  - Connect all CM folks, just to one another
  - Self-maintaining, high dynamics

Brainstorming



- Will soon see a huge explosion of different kinds of devices/services on the web
  - How do we internally compose & model these services?
  - Are we addressing the MBA community, e.g.? No.
  - Perhaps domain-specific modeling languages would be helpful for these users
    - Could this define the “anti-set”? SAP, a reference model, works pretty well for the business world. But what are the abstractions you want to come up with for the sports-specific modeling language (SSML)?
      - The world is not static; Symbian needs domain-specific tools, else it won’t work
  - MDA isn’t the way to go; UML is too generic
  - Can user compose these services herself?

Brainstorming

- Conceptual Modeling, as a discipline, offers abstractions
  - We can introduce abstractions in many domains
  - Domain-specific systems, services, models, meta-models, ontologies, ..., can help
    - Sudha votes YES on domain-specific models, especially for bioinformatics
- It will be easier to talk to users if we use <u>their</u> terms

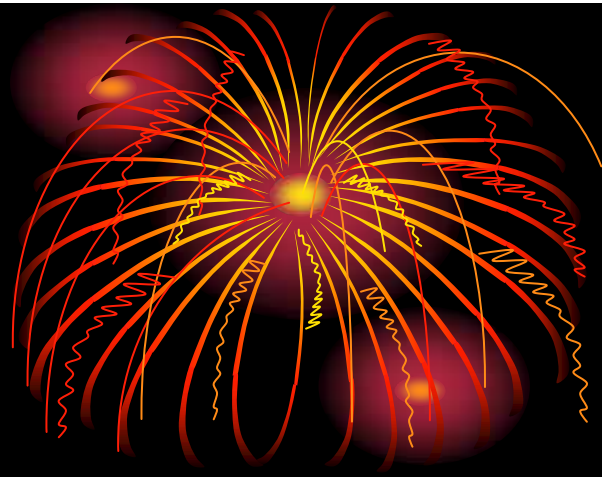
Brainstorming

- Not everyone agrees that modeling is a good idea (particularly SE community)
  - How do we overcome this problem?
  - Perhaps by creating good tools

Brainstorming

- What is the role of user profiles (esp. w.r.t. the semantic web)?
  - Can we model them?
  - Can ontologies help?
  - What does “near” mean in Atlanta, Georgia? What does it mean elsewhere?
- Well-defined domain example (oboe reeds)

Brainstorming



- Comprehension
  - Unambiguous knowledge transfer depends on shared understanding of a common model; the model is the “sine qua non” of communication.
  - The user talks in his/her own language, but must live in a world created by software designers. We need a way to reconcile the two underlying models.
  - We need to model as well as the physicists do.
- Non-software specialists don’t abstract well; our tools must operate well at different levels of abstraction

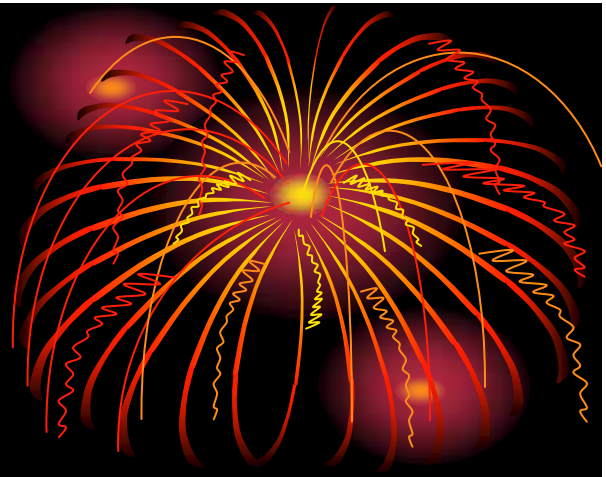
Brainstorming

- Multi-dimensional conceptual modeling
  - Data + security + superimposed functions + rights + roles + user preferences + ...
- Within 10-15 years software will be produced automatically
  - What we need are correct, high-quality CM approaches to master complexity
  - Perhaps not domain-specific modeling languages, but yes domain-specific ontologies.
  - Need to be able to map between different levels completely

Brainstorming

- There are many CM's or variations – how do you evaluate them? I don't know. How do you live with them?
  - We need methods and tools to handle different models at the same time
  - Meta-modeling/model management deserves attention

Brainstorming

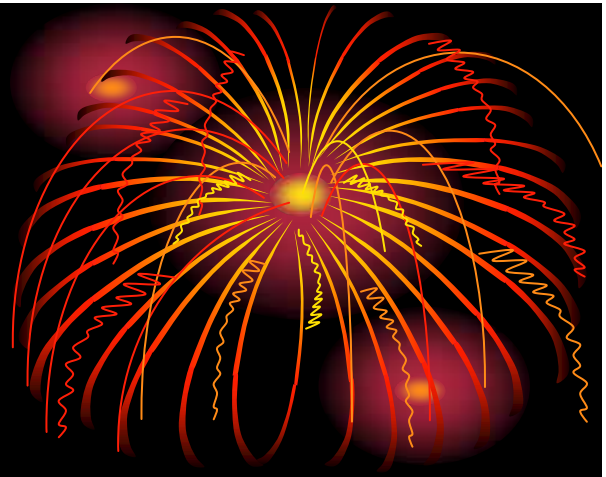


- CM of data warehouses
  - Companies are still having problems and investing resources in old data warehouses
  - Data quality/methods: no formal way to compare schemas; we need formal, objective measures
  - Security is typically an added layer later in the process; security needs to start earlier
  - P2P, GIS, Web, Semantic Web warehouses
  - MDA is the right way to design DW from CM point of view

Brainstorming

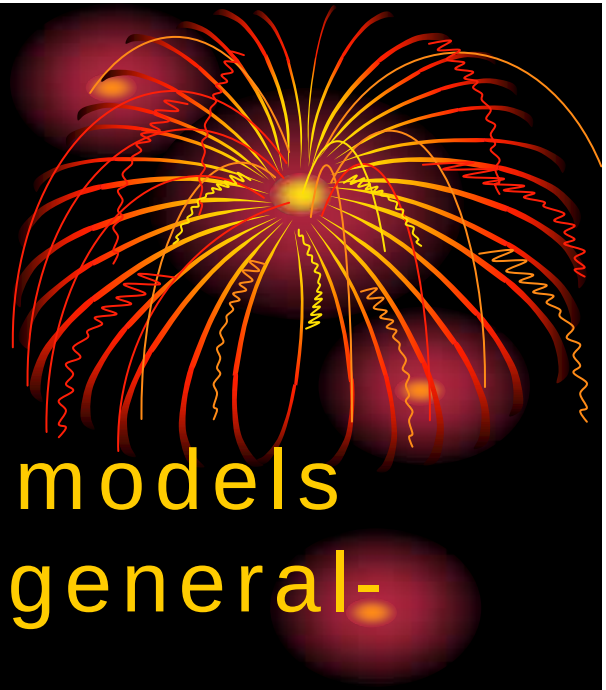
- “The devil is in the details.”
  - Programming will not disappear – you have to attend to the details.
  - Can you express the details in some other way, e.g. in the CM?

Brainstorming



- MIS perspective
  - Would like to see large-scale studies to evaluate the utility of various models – don't just come up with proof of correctness, completeness, etc. We need empirical evaluations too. Consider productivity increase, error-rate decrease, ...

Brainstorming



- Would data warehouse models use domain-specific or general-purpose models?
- A posteriori approaches to converting the web to the semantic web have merit

Brainstorming

- Look into the literature for “pragmatics” (historically: semantics, syntax, pragmatics)
  - How can we include pragmatics in our CM’s?
  - Users may have restricted scope, attention, etc.
  - Schema redundancy in SAP is 4.5x

Brainstorming

- The point of domain-specific models is the reduction of complexity
  - The end-user really only cares about fake screens and maybe a prototype
  - Let’s not look too far away from a general model

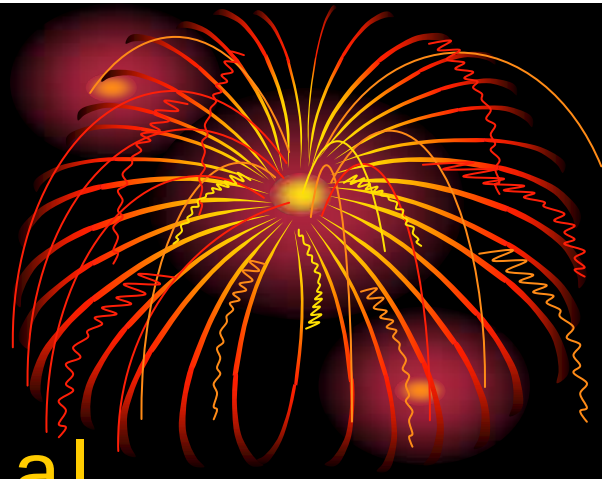
Brainstorming

- Ontologies will make the web semantic
- Ontologies are in the hands of the AI people
  - They don’t know what a user is
  - We need to enhance CM with reasoning abilities
- The web is basically a P2P system
  - Emergent semantics – meaning arises from what is there, not imposed from elsewhere
  - Emergent semantics is potentially a hot topic
  - Context-aware databases are absolutely something we need to provide, and soon

Brainstorming

- “Find a nearby friendly airport that can land an F-15.”
  - We don’t know how to effectively connect semantics from different domains
- Knowledge is often insufficient or incomplete

Brainstorming

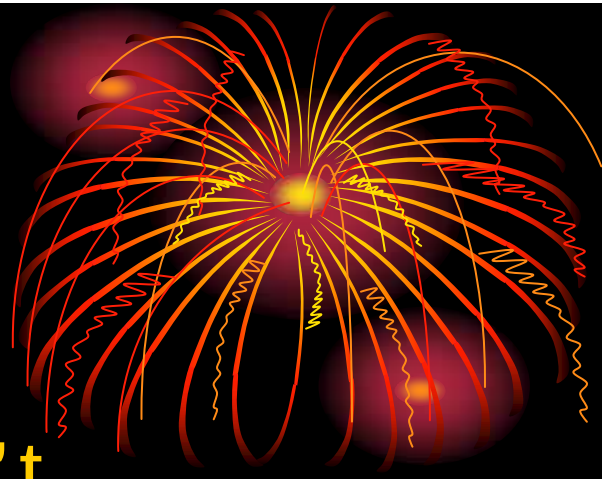


- Don’t think of conceptual modeling as just something that happens in the initial modeling phase
  - Can use CM for formative evaluation (in contrast to summative evaluation; cf. HCI)
  - Get user feedback in our evaluation

Brainstorming

- Domain-specific programming language tries to formalize small set of specific operations in order to formally verify programs in the language
  - Formal verification easily becomes intractible

Brainstorming

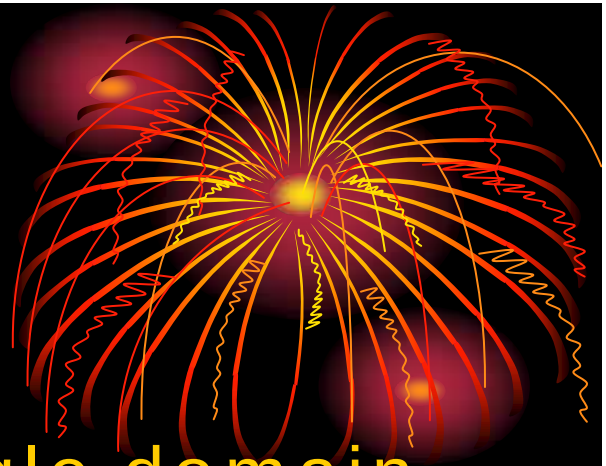


- Yes, programming won't disappear, but it will have less importance
- We need to adopt industrial processes in our software development
  - Standardized, high-quality components composed to make products

Brainstorming

- With model compilers, programming won't disappear – it moves "up the food chain"
  - Modeling becomes the system development activity
  - A challenge is to provide general CM's that can be applied in specific domains as needed
- Instead of bottom-up, we need some top-down approaches

Brainstorming



- You can study CM in a single domain or in multiple domains
- When domains are independent, CM's are independent
- When domains overlap, we need to be sure our conceptualizations appropriately treat similar concepts the same way in both domains

Brainstorming

- CM is only conceptual if you forget 100% about implementation
- If you have a CM you're happy with, you can add arbitrary amounts of complexity
  - You can go forever with multi-dimensional models, e.g., as long as you observe the principle of orthogonality; otherwise you get into a mess.

Panel Participants

- Paulo Atzeni (Università Roma Tre)
- Wesley Chu (UCLA)
- Lois Delcambre (Portland State University)
- David W. Embley (Brigham Young University) – moderator
- Burkhard Freitag (Passau University)
- Stephen W. Liddle (Brigham Young University) – scribe
- Heinrich Mayr (University of Klagenfurt)
- Erich Neuhold (Frauenhofer IPSI)
- Antoni Olivé (Polytechnic University of Catalonia)
- Óscar Pastor (Polytechnic University of Valencia)
- Sandeep Purao (Penn State University)
- Sudha Ram (Arizona University)
- Matti Rossi (Helsinki School of Economics, Information & System Science)
- Arne Sølvberg (Norwegian University of Science & Technology)
- Stefano Spaccapietria (EPFL Lausanne)
- Veda Storey (Georgia State University)
- Bernhard Thalheim (Kiel University)
- Juan Trujillio (University of Alicante)
- Gerhard Weikum (Max Plank Institute of Computer Science, Saarbrücken)