

Using Software Process

to Support Learning Usisi



# Overview



## § Software process initiatives

- ❖ **from process to a repository of best practices**

## § The BORE system and methodology

- ❖ **tailoring process to project needs**
- ❖ **delivering resources and best practices**

## § Conclusions and future work

- ❖ **ISC Software Methodology at Goddard**

# Software Process Initiatives

## § High-level activities

- ❖ **must address the least common denominator**
- ❖ **complexity is an issue**
  - ...when viewed as a monolithic document
- ❖ **little support for how the activity should be carried out**

## § Often seen as a definition effort

- ❖ **little thought on using feedback to refine the process**

## § Disseminating best practices

- ❖ **use process to show how others have performed the activities**
- ❖ **building an organization-specific knowledge base**

# Overall Approach

## § Process tailoring

- ❖ **define applicability rules for activities**
- ❖ **“if there are significant technical risks”**
  - **“document the risks, perform prototyping activities”**

## § Flexible software process

- ❖ **SDM as a repository of best practices**
- ❖ **more detailed knowledge than SDMs**
  - **capture more than the least common denominator**
- ❖ **supporting process diversity**
  - **tailor SDM to specific project needs**
- ❖ **document project-specific issues**
- ❖ **use assigned tasks to disseminate knowledge**

# BORE



## § Building a Organizational Repository of Experiences

- ❖ **using process to organize and manage knowledge**
- ❖ **capturing the context for using different techniques**

## § Case-based repository

- ❖ **hierarchical representation of project activities**
- ❖ **instantiation of activities are a cases**
  - **problem and solution statement**
  - **document context-specific information**

# BORE Project Interface

The screenshot displays the BORE Project Interface. On the left, the 'Case Manager' window shows a hierarchical tree structure for the 'Goddard ISC demo' project. The tree includes sections for 'Overall Project Parameters', '1.0 Requirements Activities', '1.1 System Concept Definition', '1.2 Requirements Definition', '1.4 Identify Hardware and Software to be Procured', '2.0 Design Activities', '2.3 Preliminary design', and '2.4 Detailed Design'. The 'Hold a Software/System Concept Review (SCR)' task is highlighted in red.

On the right, a Netscape browser window titled 'Hold a Software/System Concept Review (SCR) - Netscape' displays the details for this task. The window has tabs for 'Overview', 'Edit', 'Resources', and 'Options'. The 'Overview' tab is active, showing the following information:

**CASE NAME:**  
Hold a Software/System Concept Review (SCR)

**DESCRIPTION:**  
It is recommended that the Team hold a System Concept Review (SCR), or participate in a higher level Project SCR. The purpose of the SCR is to give customers and other interested parties an opportunity to examine and influence the proposed system and operations concept before detailed requirements are written. If an SCR is held at the project level the Team Lead shall collect requests for action (RFAs) and shall track RFA responses and resolutions.

**SCR Topics:**

- Top level requirements
- High level, "ideal," architecture, identification of components to be reused
- Identification of major external interfaces
- Operations concepts for major modes of operation.

**Tailoring Note (reuse:)**  
For a system with significant reuse, and where both the development team and the customers have experience with the system to be reused, the focus of the review is on similarities and differences between the system planned for reuse and the target system. The review helps to determine if the gaps have been properly identified and addressed. [Scott Henninger : 11/15/2000]

At the bottom of the Netscape window, there is a field for 'Owners' with the value 'Scott Henninger : 11/28/2000', a 'Status' field with the value 'Open/Critical', and a 'Close' button.

# BORE Domains



## § Domains in BORE

- ❖ **define activities for projects**
- ❖ **define when a given activity applies**

## § Domain Cases

- ❖ **case-based paradigm**
  - “principles” contain general rules or knowledge
  - cases specialize the principles
- ❖ **domain cases play the role of principles**
- ❖ **projects belong to a domain**
  - domain defines all cases for that domain

# Domain Rules

---

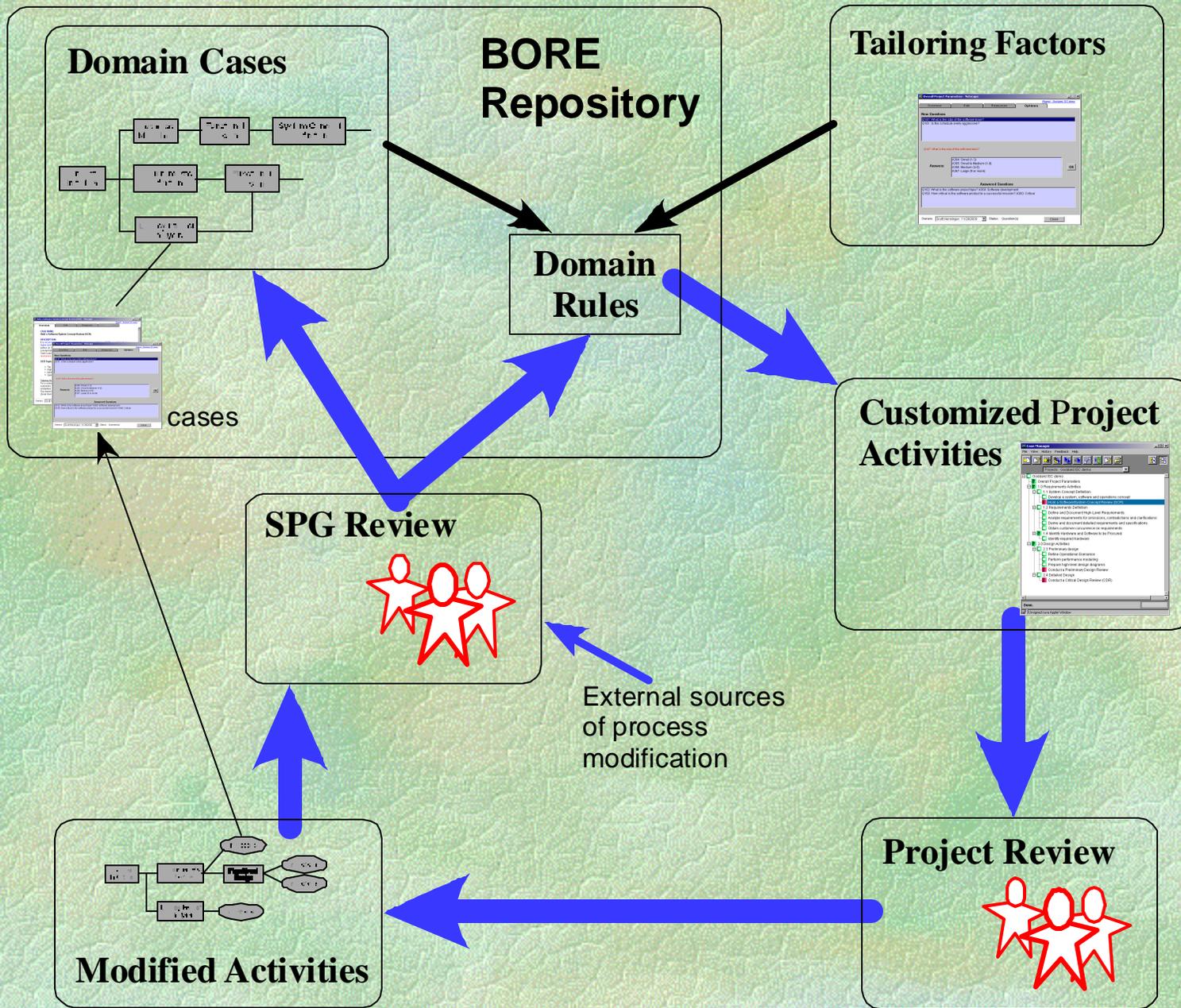
## § Tailoring to specific project needs

- ❖ **capture context in rules**
  - **under these conditions, assign this task**
  - **can be as high-level as an SDM, or very detailed**
- ❖ **really capturing project decisions, requirements**

## § Rule-based engine

- ❖ **preconditions: question/answer pairs**
- ❖ **actions: assign tasks, other actions**

# BORE Methodology



# Tailoring to Project Needs

§ Tailor the SDM to specific project needs

- ❖ **deviation process when current rules/tasks don't apply**
  - i.e. a “breakdown” requiring new actions
  - new domain activity to describe the task
  - rules for tailoring (encode rationale for the deviation)
- ❖ **allow subsequent projects to use tailored processes**
  - i.e. set a precedent
  - match problem to specific context
- ❖ **allow the SDM to grow “organically”**
  - ...as new problems are encountered, technology evolves, etc.

§ Beyond the expert system paradigm

- ❖ **rules as a resource for human action**
- ❖ **collaborative creation of rules**

# Iterative Disclosure of Detail

- § Any case can have process decisions
  - ❖ decisions define increasingly detailed project tasks

Overall Project Parameters - Netscape

Project : [Goddard ISC demo](#)

Overview Edit Resources Options

**New Questions**

Q147: What is the size of the software team?  
Q161: Is the schedule overly aggressive?

Q147: What is the size of the software team?

**Answers:**

A364: Small (1-3)  
A365: Small to Medium (1-8)  
A366: Medium (4-8)  
A367: Large (9 or more)

**Answered Questions**

Q162: What is the software project type? A368: Software development  
Q160: How critical is the software product to a successful mission? A360: Critical

Owners:  Status:

# Current System Status

## § BORE prototype

- ❖ <http://cse-ferg41.unl.edu/bore.html>
  - ==> Bore v. 3.2
- ❖ log in as 'guest'

## § Just a prototype...

- ❖ **documentation is lacking**
- ❖ **currently only works on Communicator**
- ❖ **no application security**
- ❖ **current rule engine, interface need work**
- ❖ **document management needs Web-based upload**

# Open Questions

---

§ Feasibility of the approach

- ❖ **increase or decrease documentation burden?**
- ❖ **need for rule-base system expertise**

§ Flexible process or encoded inertia?

- ❖ **need a disciplined, yet flexible process**

§ Expert system paradigm revisited?

- ❖ **rules as a resource**
- ❖ **sustained knowledge acquisition**

§ Process representation is simplistic

- ❖ **good or bad?**
- ❖ **some form of default dependency representation probably necessary**

# Future Work



## § Goddard ISC Software methodology

- ❖ **current focus on ISO 9000 compliance**
- ❖ **matrix representation (Schultz et al.)**
- ❖ **using BORE to encode matrix and tailoring factors**
- ❖ **pilot study using BORE**
  - **use in small project not on critical path**

## § Review Process

- ❖ **elaboration on how reviews are conducted, etc.**
- ❖ **guidelines for how and when knowledge should evolve**

## § Collecting measurements

- ❖ **use cases to collect data in a single place**
- ❖ **define measurement points in process definition**





## § Capturing detailed information

- ❖ **case-based architecture**
- ❖ **software process defined by activities**
- ❖ **activities are assigned to projects**
  - **projects document, adapt the activities to their needs**
  - **i.e. provide context-specific information**
- ❖ **project cases cross-referenced with the activities**
  - **easy reference to project experiences**