

2009 AF SOA Developer's Conference

Integrity - Service - Excellence



**Supporting Collaboration
& Knowledge Sharing in
Software Development**

**Scott Raley
643 ELSS/EIPA**

Date: 21 May 2009

554 Wired for War



Knowledge Sharing & Collaboration



What are we really talking about?

- A range of practices and technologies used in an organization to identify, create, represent, distribute and enable adoption of artifacts, lessons learned, best practices, and knowledge

Developer Collaboration

- Few software developers have all the knowledge needed for developing today's complex software systems so collaboration has become an essential aspect of software development and implementation



Knowledge Sharing

- Encompasses the transfer of knowledge among software developers and the collaborative creation of new knowledge that is needed for the development/implementation/maintenance of software systems
 - Tacit knowledge – subjective, experiential learning – hard to document hard to transfer/teach/learn - involves a lot of human interpretation
 - Explicit knowledge - objective, rational, technical - easily documented - easily transferred/taught/learned



Drivers



Software development/implementation is a very knowledge intensive activity in which trial-and-error is a dominant technique in resolving problems

- Software development requires extensive knowledge from a wide range of domains and sources
- Developers can greatly benefit from reusing positive and negative experiences that other developers have had in the past
- Helps retain development knowledge and expertise in face of retirements, transfers, etc.

With the expansion of software outsourcing and service related development, geographically distributed software development is rapidly becoming a dominant development model

- A new personalized, context-sensitive and decentralized concept for sharing software development knowledge that can be seamlessly integrated into our software development /implementation environment is required





Challenges



Organizational Boundaries

- Often collaboration practices are organizational proprietary
 - Each organization determines level & manner of involvement
 - Multiple standards & practices (coding standards, engineering Processes, etc)
 - Funding issues
 - Can lead to silo mentality

• Contextual Limitations

- Information that seems to be irrelevant for resolving a given problem from the point of view of a developer could be very relevant for another developer, due to his/her background knowledge or the given working context
- The problem is not only to find and store experience, but to represent it in a highly reusable form





What Do We Need



Capabilities

- **Communication**
 - Enable people to connect to other people
- **Collaboration**
 - Enable people to work together
- **Storage**
 - Store and maintain the knowledge - who has the knowledge – who can access – how to access

Supporting Technologies

- **Synchronous**
 - Instant messaging
 - Video/audio conferencing
 - IRC (chat room)
 - Application/whiteboard sharing
- **Asynchronous**
 - Mailing-lists
 - Message/Discussion boards
 - Wiki
 - E-mail
 - Shared Calendars
 - Structured & unstructured data repositories
 - Process/workflow support





AFKN Communities of Practice



Knowledge Now is a web-based Air Force Materiel Command (AFMC) system designed to support virtual collaboration, people networks, and on-the-job learning

The concept of AFKN revolves around Communities of Practice (CoPs)

Each CoP residing within AFKN is a self-contained, virtual community that represents a particular segment of the Air Force workforce

AFKN CoP collaborative features:

- Community Calendar
- Discussion Forums
- Document Management
- Tell a Friend
- Wisdom Exchange





AFKN CoPs

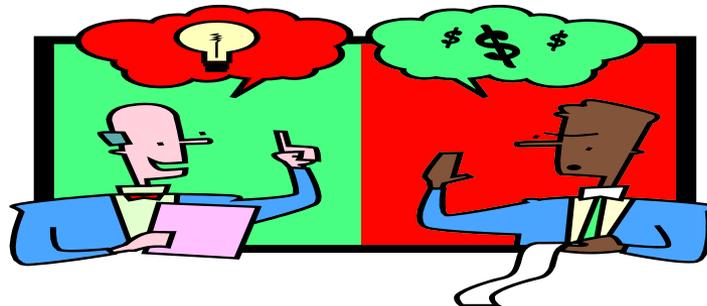


Pros

- Existing solution – no cost for use/implementation
- Some knowledge sharing capabilities – Discussion boards, Wiki
- Document management capabilities

Cons

- Primarily a knowledge management solution
 - Not geared towards collaborative development
- Locked in to tools/processes/solutions provided by AFKN (Wikis, discussion forums, etc)

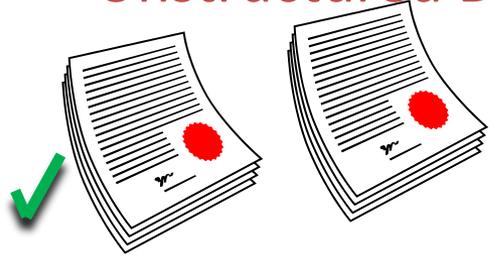




AFMC Enterprise Information Management (EIM) System



Unstructured Data



Document Management

Version control

Check in/out

Records Management



"Google like" Search

Process Automation (Workflow)



- ✓ Production Ready
- ✓ Future Capability

Knowledge Management

Blogs

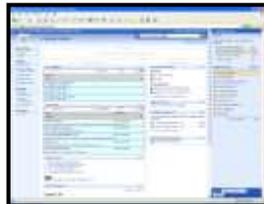
Wikis

Ask an Expert



EIM Capabilities

Team Collaboration



Management Dashboards



Structured Data



Other Standard Business Systems (ProSight, Powersteering, etc.)

External Data Systems



EIM SharePoint Sites

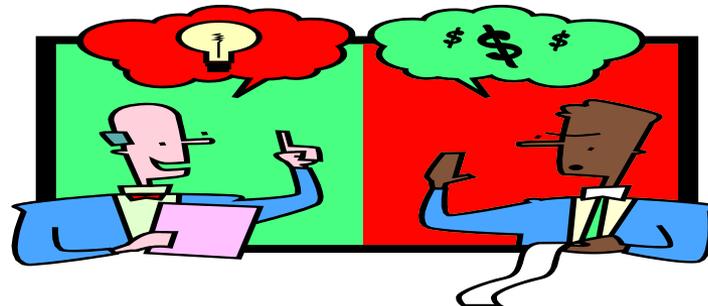


Pros

- Existing solution – no cost for use/implementation
- Strong knowledge management capabilities – Wiki, blogs, ask-an-expert
- Some team collaboration support – workflows – shared calendars
 - Projected to have dashboard and more extensive workflow support

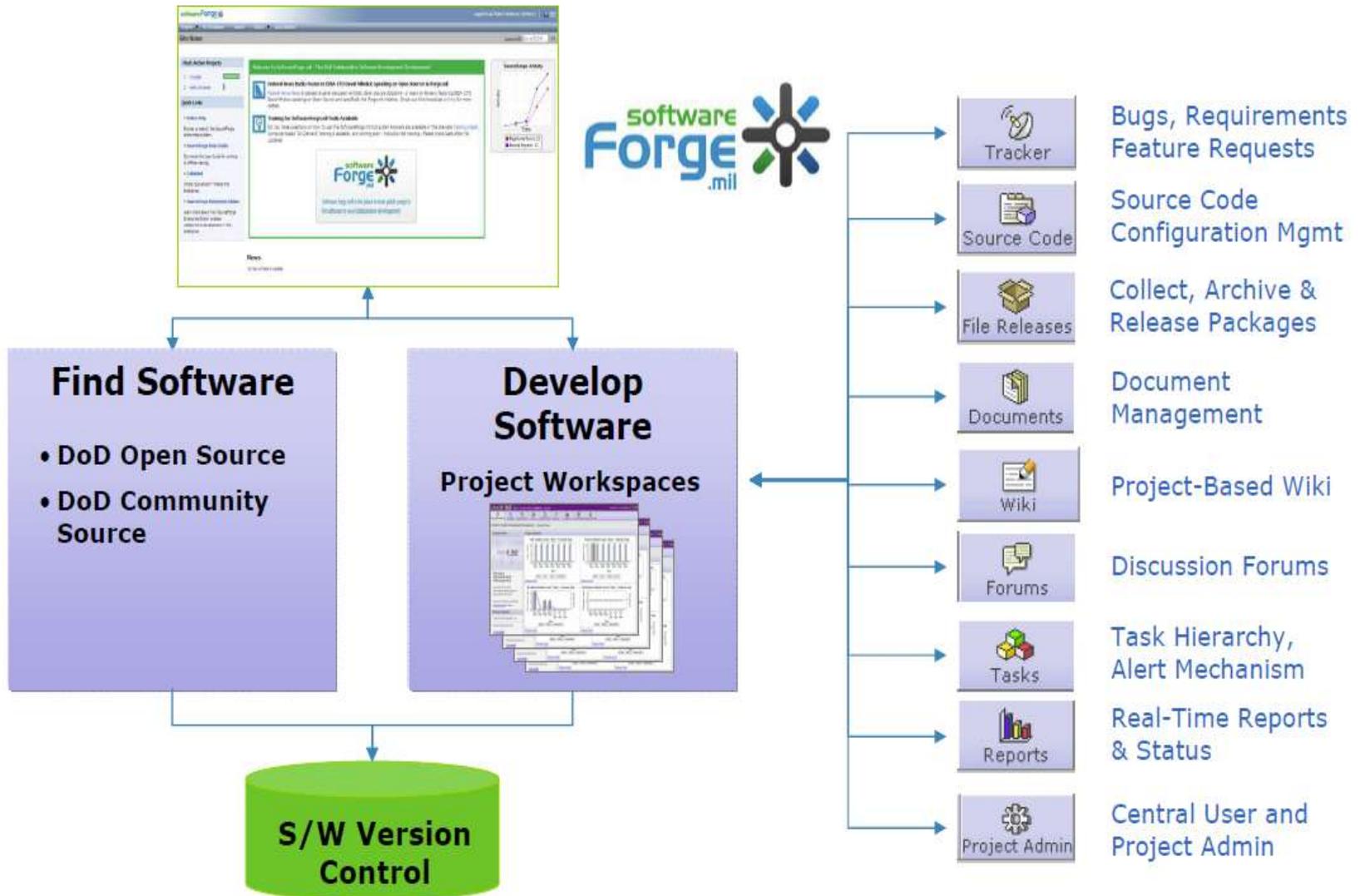
Cons

- Currently limited to AFMC users
 - Scheduled for .mil access in June 2009 - .com & .edu later
- Primarily a knowledge management solution
 - Not geared towards collaborative development
- SharePoint enterprise solution
 - Locked in to tools/processes/solutions provided by EIM
- Potential for storage issues





SoftwareForge.mil





Software.forge.mil

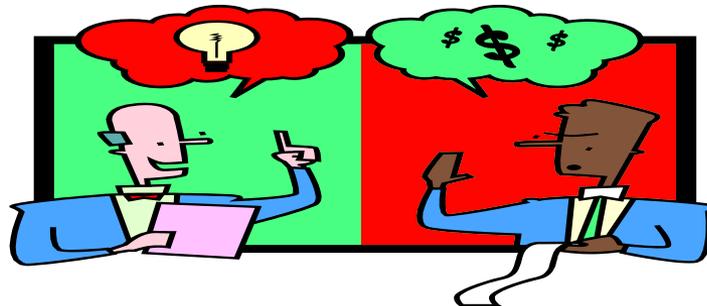


Pros

- Existing solution – no cost for use/implementation
- Strong team development and collaboration support
 - Supported by CollabNet platform
- Full integrated suite of code, build, test & collaboration tools

Cons

- Geared mostly towards team/project development collaboration
 - Does have some knowledge management support
- Access is limited U.S. military, DoD government civilians and DoD contractors – CAC access – no access for private industry
- Fairly new – support/standards/processes potentially in flux





Other Solutions

IBM Jazz

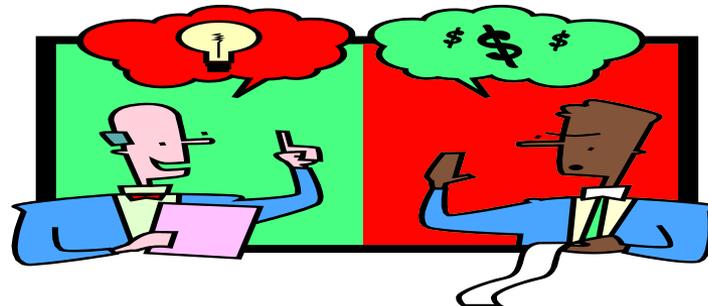
- End-to-end team collaboration and governance across all disciplines of the development lifecycle

Web-Based Solutions

- Bricolage
- Dimdim
- DotNetNuke
- eGroupWare
- Group-Office
- Horde
- OpenGoo
- phpGroupWare
- ProjectPier
- Simple Groupware
- SlashCode
- TikiWiki CMS/Groupware
- Tine 2.0
- Wiggio

Project Collaboration Solutions

- dotProject
- eGroupWare
- Fle3
- Horde
- JavaForge
- Mindquarry
- OpenGoo
- phpGroupWare
- Plone
- project.net
- SharpForge
- Trac In



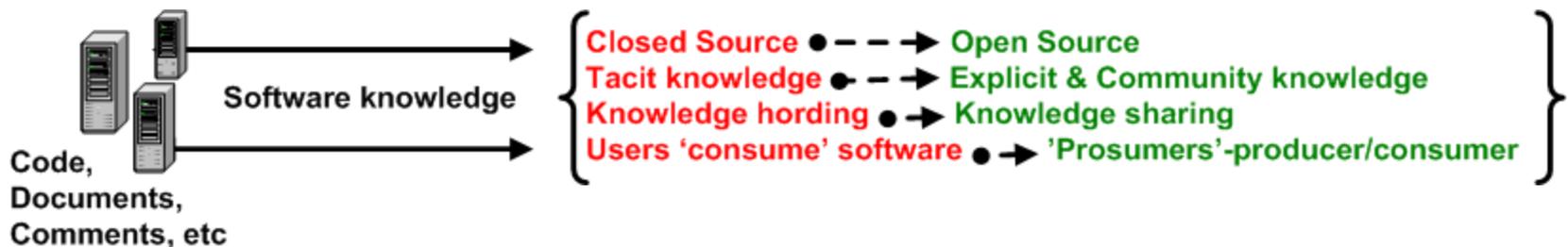


Way Ahead - General



Make reusable components, developer expertise, and knowledge easily available across organizations regardless of location or organizational constraints

- Foster a collaboration culture
 - Identify and remove barriers to adoption
 - Reward innovation
- Manage by letting it happen
- Provide organizational support and skills
- Highlight and promote professional and organizational benefits
- Set aside play time
- Expand out past our program and organizational 'silos of excellence'





Way Ahead - Specific

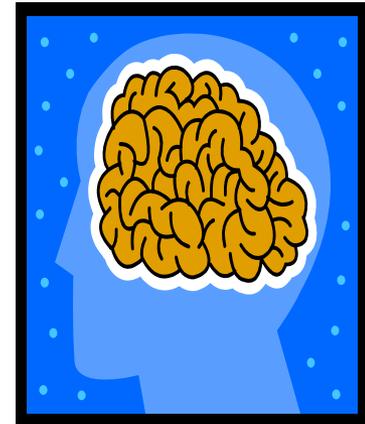


Stand up a collaborative and information sharing environment across DoD & private industry

- Accessible to DoD and private industry
- Use open-source development/implementation approach
 - For developers, by developers

Set up regular “open” brainstorming sessions to establish:

- Processes/Practices
 - Group notifications/communications
 - Publishing and discovery
 - Tiered access
 - Support roles (moderators, SMEs, etc)
- Technical solutions
 - Evaluate existing solutions
 - Submit candidate solutions
 - Select & implement





How Can You Help?



First brainstorming session is scheduled for 1 June 09 @ Gunter

- Dial-in is available
- Grab a flyer – first session and contact info

Get involved

- Developers, managers, vendors
- Enterprise benefits are only realized if everyone in the enterprise helps
- Only limitations are self-imposed





Wrap-Up



Developer Collaboration & Knowledge Sharing

- The definition is clear
- The concepts are clear
- The challenges are
 - Clear
 - Surmountable
- The benefits are clear (and can be huge)
- The tools and technologies are viable and available (mostly)



The challenges of software development are certainly not going to go away, for we as an industry are continually being driven to do more with less. Methods and processes help; so do languages, frameworks, and tools. However, software development is ultimately a human endeavor, and as such it's ultimately the efforts of the software development team that enable us to deliver quality systems in a predictable and sustainable fashion.” (Grady Booch)



Questions

