[McConnell 1996] Steve McConnell, *Rapid Development: Taming Wild Software Schedules*, Microsoft Press, 1996, pp. 449-463. ISBN 1-55615-900-5

Joint Application Development (JAD)

JAD is a structured methodology for

- Requirements gathering and negotiations
- User interface design

Venu: Intense workshops offsite

JAD leverages group dynamics, extensive use of visual aids, WYSIWYG documentation, and an organized rational process to gather requirements in a short time.

Key participants must participate full time (no observers allowed)

- End users
- Executives <u>"Joint" = together participants design product concept</u>.
- Developers

Goal:

• Specify the systems details from the business perspective (<u>NOT technical perspective</u>)

Results:

- shortens time spent in requirements phase
- American Airlines, Texas Instruments and IBM reported reductions of 15 to 35% in 1985 CNA Insurance Company cut requirement effot by almost 70%. [Ruth 1985]. Normal reduction in requirements efforts 20% to 60%, in total effort 20% to 60% [August 1991] Requirements gathering takes between 10% to 30% of elapsed time on typical project Can expect a total development time reduction of 5 to 15% [Boehm 1981]
- captures requirements better
- reduces requirements changes, and helps get the requirements right the first time
- helps get the user interface right the first time, higher quality
- commits top executives to the software planning process, shortening the product-approval cycle.
- eliminates features of questionable value, reducing the size of the product and the development time
- reduces organizational infighting by exposing hidden agendas, conflicting needs, and politics that can cripple a project

Has been used for development of:

- business systems
- shrink wrap software
- systems software
- client-server software
- vertical market software

Success depends on:

- Effective leadership of the JAD sessions
- Participation by key end-users, executives and developers, and
- Achieving group synergy during JAD sessions

JAD Leader

• specially trained in the JAD structured process to facilitate reaching consensus and run efficient meetings

Efficacy:

- Potential reduction from nominal schedule:
- Improvement in progress visibility:
- Effect on schedule risk
- Chance of first-time success:
- Chance of long-term success:

Facilities:

- Located offsite, ideally in hotel or conference center
- Visual aid support
- Computer support
- Copy machine, Polaroid camera to record whiteboard contents
- Notepads, pens
- Name cards (if needed)
- Drinks and refreshments

Good Fair Decreased Risk Good Excellent

Roles of participants (1 of 2)

• Session leader

- Excellent communication & mediation skills, deals effectively with
 - Political disputes
 - Power struggles
 - Personality clashes
 - Controlling group including high-level executives
 - Encourages quiet participants and prevents strong personalities from dominating sessions
- Must be impartial, keep an open mind and control controversies

• Executive sponsor

- has financial responsibility for the system
- Makes the go/no-go decision

• End-user representative(s)

- Has authority to make binding decisions about the program
- Must be good communicator

Roles of participants (2 of 2)

• Developer

- More than one developer should attend
- Answers questions about feasibility of proposed features and costs
- Role is to provide information, not make judgements
- Must learn the system during JAD planning and design sessions to prepare them to write the specifications

• Scribe

- From the software development department to record what happens during the session.
- Active participant, asking for clarification and pointing out inconsistencies day by day.

• Specialists

- Invited as needed to provide any special expertise required
- Exception to rule that all participants must be present at all times
- Resource rather than a member of the group

Major Risks:

- Unrealistic productivity expectations following the JAD sessions
- Premature, inaccurate estimates of remaining work following JAD sessions

Averting problems:

- If you can't get key people to participate, don't do the JAD at all
- Don't allow observers to attend because they can't keep their mouth shut. If they are forced on you, make them attend the same training sessions as the participants
- Too many participants keep the JAD team from jelling. Accept no more than 8 people in the JAD team. If organization is skilled at JAD can be successful up to 15 people

Major interactions and trade-offs

- Works best when combined with an incremental-development lifecycle model
- Can be combined with
 - rapid-development languages and
 - prototyping tools

Two main phases:

- JAD-Planning Phase (also called Joint Requirements Planning (JRP))
 - Goal: map broad capabilities of system from the business perspective
 - Outcomes:
 - System's goals
 - Preliminary effort and schedule estimates
 - Decision whether to continue
- JAD-Design Phase
 - Goal: to create user-level design of the software
 - Detailed user-interface design
 - A database schema (if appropriate)
 - Refined budget and schedule estimates
 - Uses prototyping extensively
 - Does NOT focus on the functional design of the system
- Subphases:
 - *Customization*: Tailor method to project, 1 10 days
 - Session: All parties meet, lasts from 1 10 days
 - *Wrap-up:* Notes and visual aids converted into formal documents. Takes 3 5 days.



JAD Analysis Session:

orientation

• define *high-level requirements*

• map out system, identify business needs system will address, system objectives, anticipated benefits, list of possible system functions, rough prioritization of system functions, and strategic and future considerations

• limiting the system *scope*

- decisions on what NOT to include is as important as what the system will include
- identify and *estimate* JAD design phase(s)
- identify the JAD *design participants*
- agree on *schedule* of the JAD design phases
- document the *issues and considerations* -- options and reasons

JAD Analysis Wrap-up Session Document

- List of system **objectives**,
 - including strategic and future considerations

• **Details** of

- possible system functions,
- business needs that each function addresses,
- benefits of each function,
- estimate of the return on investment of each function,
- a rough prioritization of each function
- Limitation on the system's scope,
 - including a list of functions that the system will NOT include
- List of **interfaces** to other systems
- List of **unresolved issues** during JAD session,
 - including name of the issue, the person responsible, and the promised resolution date
- Plan for what happens next,
 - including identification of follow-on JAD design phases, JAD design participants, JAD design schedules and rough target dates for implementation.

JAD Design Session:

- orientation
- *review* JAD planning requirements and scope
- develop *workflow diagrams* show how work will be done with new system
- develop workflow description in words
- design the screen layouts and report formats
 - extensive use of interactive prototypes created by developers and end-users during and between meetings
- specify *processing requirements*
 - specify data volumes, rates, audit requirements, security requirements, ...
- define *interface requirements*
 - specify systems that the new system must interface to
- identify *system data* groups and functions
 - map out major data structures and relationships. If system is database-oriented create a normalized database schema
- document the **issues** and consideration
 - Record the options considered and reasons issues were resolved how they were

Design Wrap-up Session Document

- Complete documents
- Complete the prototype
- Review documents, and prototype by participants
- Present result to the executive sponsor, including
 - a summary of the design session,
 - the JAD design,
 - preliminary target implementation dates, and
 - the project's current status

References:

[Wood 1995] Jane Wood, and Denise Silver, *Joint Application Development*, 2^{nd} Ed.