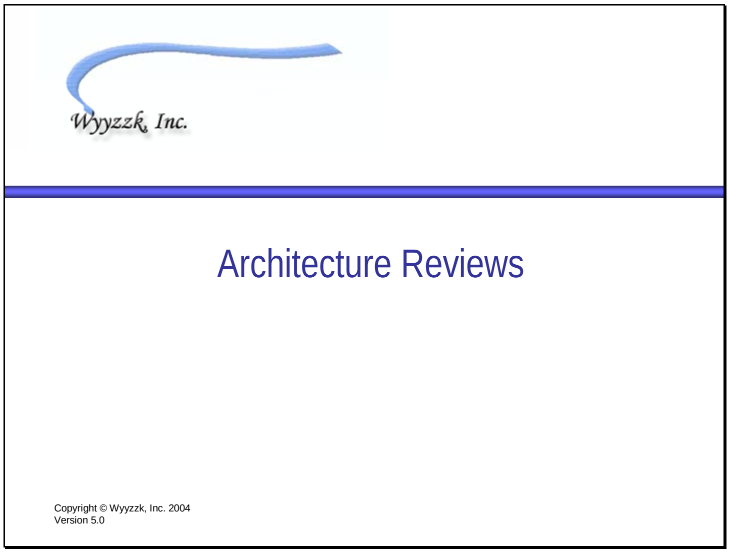
Slide 1

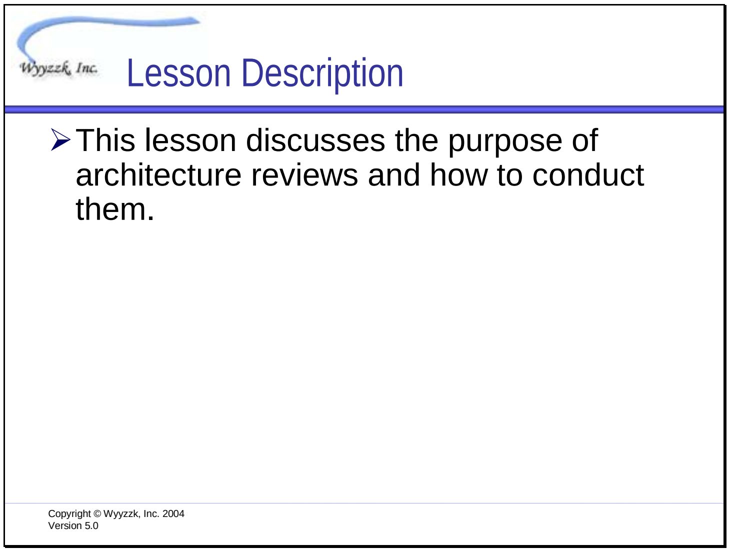


Wyyzzk, Inc.

Architecture Reviews

Version 5.0

Slide 2



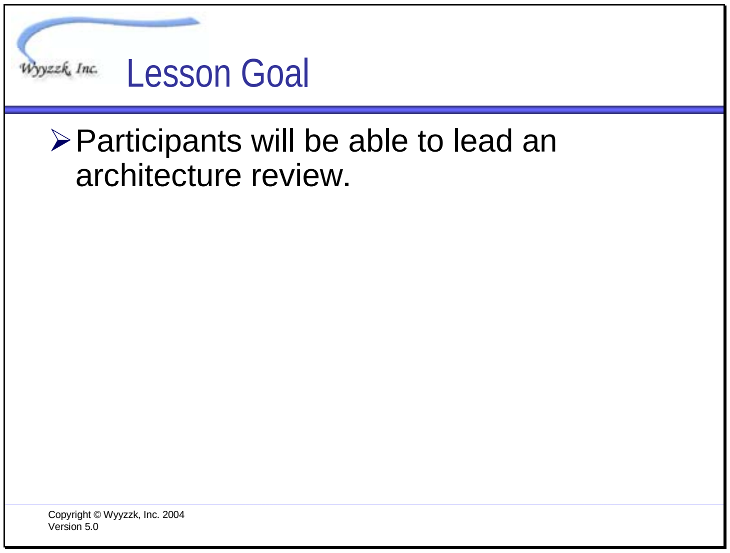
Wyyzzk, Inc.

Lesson Description

- This lesson discusses the purpose of architecture reviews and how to conduct them.

Version 5.0

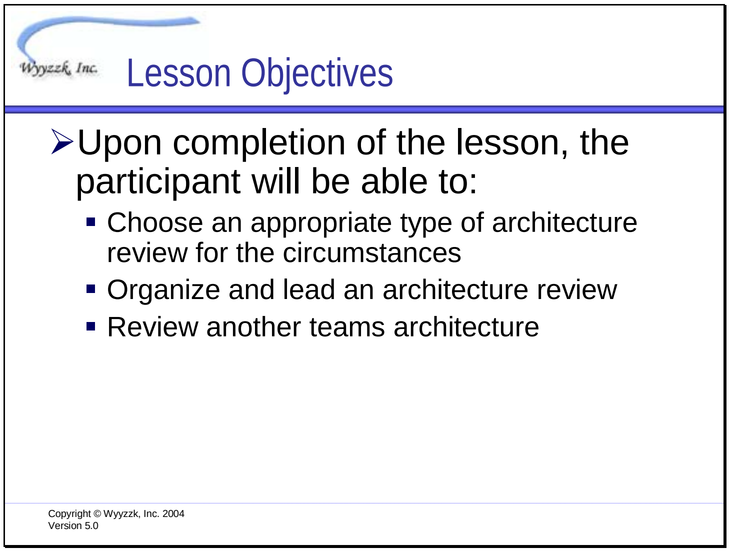
Slide 3



Lesson Goal

- Participants will be able to lead an architecture review.

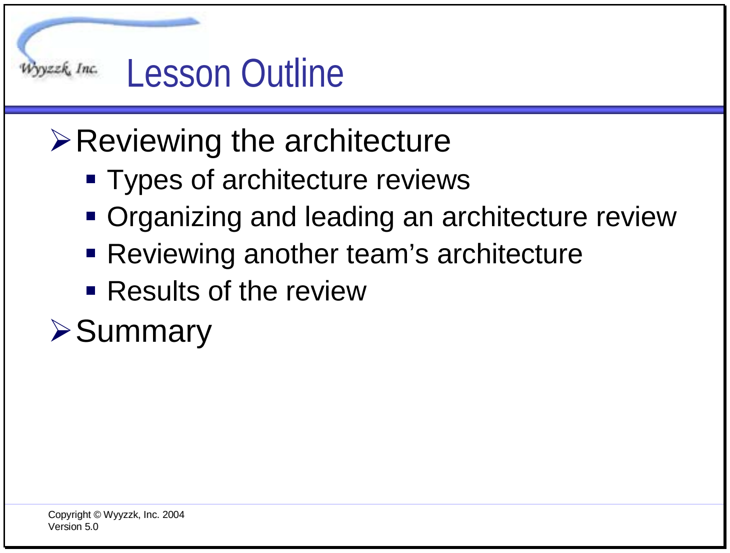
Slide 4



Lesson Objectives

- Upon completion of the lesson, the participant will be able to:
  - Choose an appropriate type of architecture review for the circumstances
  - Organize and lead an architecture review
  - Review another teams architecture

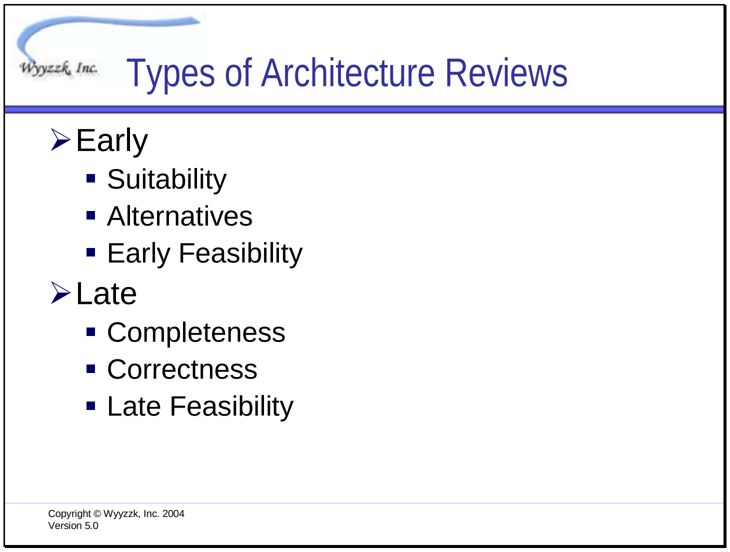
Slide 5



Lesson Outline

- Reviewing the architecture
  - Types of architecture reviews
  - Organizing and leading an architecture review
  - Reviewing another team’s architecture
  - Results of the review
- Summary

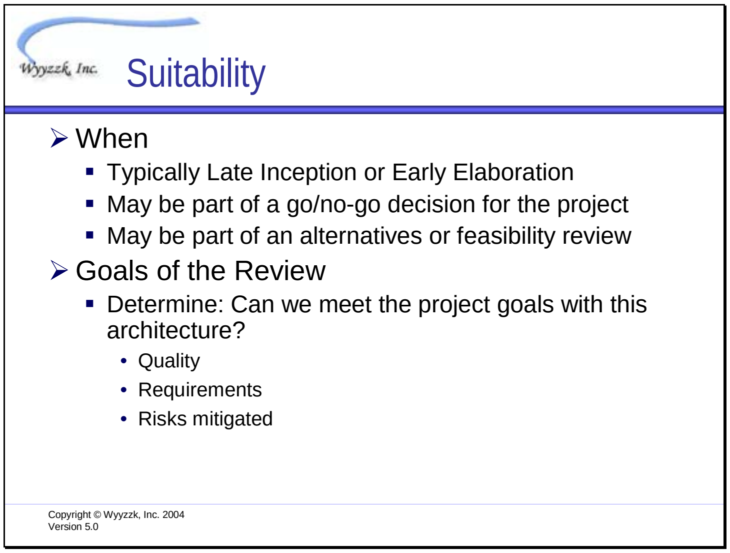
Slide 6



Types of Architecture Reviews

- Early
  - Suitability
  - Alternatives
  - Early Feasibility
- Late
  - Completeness
  - Correctness
  - Late Feasibility

Slide 7

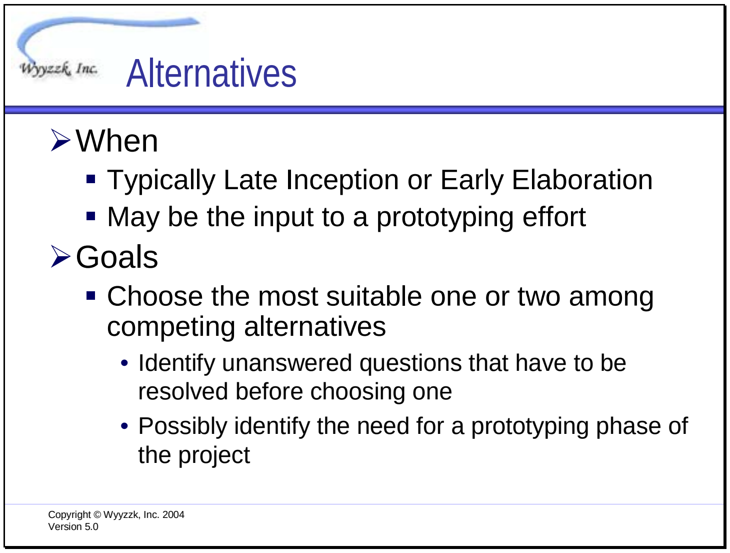


Suitability

- When
  - Typically Late Inception or Early Elaboration
  - May be part of a go/no-go decision for the project
  - May be part of an alternatives or feasibility review
- Goals of the Review
  - Determine: Can we meet the project goals with this architecture?
    - Quality
    - Requirements
    - Risks mitigated

In this review, we are just worried about whether the architecture is suitable for the project. Does it handle the important requirements, quality attributes (performance, security, etc.), and risks? This may also be combined with a feasibility review where we determine if the architecture that is suitable is a feasible one for the company, or an alternatives review where we contrast the suitability of more than one architecture to choose the best for our project.

Slide 8



Alternatives

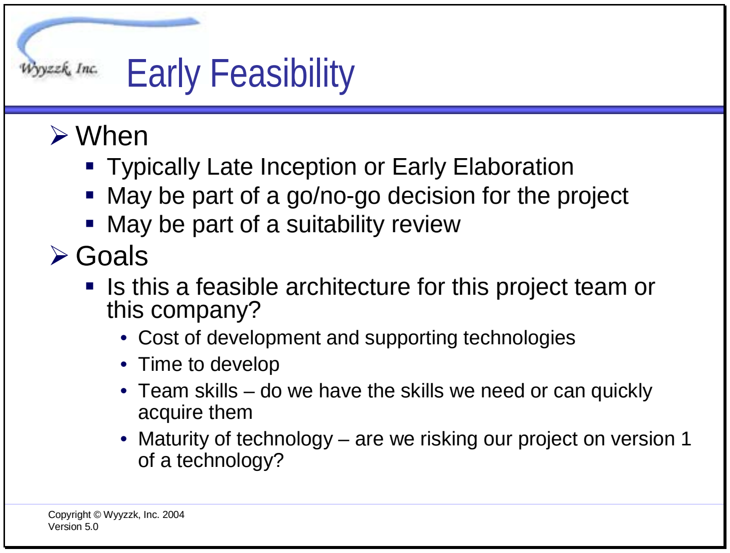
- When
  - Typically Late Inception or Early Elaboration
  - May be the input to a prototyping effort
- Goals
  - Choose the most suitable one or two among competing alternatives
    - Identify unanswered questions that have to be resolved before choosing one
    - Possibly identify the need for a prototyping phase of the project

Sometimes you cannot pick one architecture early in the project. You may need to get answers to some questions before the best architecture can be determined. To get those answers typically requires a prototyping phase, where you write code to run some tests to answer the question. (If the questions can be answered without writing code, then they should have already been answered by the architecture team.)

These kinds of questions often end up in the area of performance (how fast is that machine in our environment, how many users do we typically have, what is the peak load on the system, which is the best hardware configuration to achieve our performance goals). I have known companies to actually borrow hardware from vendors such as Sun, Microsoft, and IBM in order to test out which is the best configuration for them. Oracle has a testing lab for customers to use to try out their ideas on Oracle equipment. This is especially true when some new hardware has come out that does not have a lot of benchmarks already.

Prototyping should always be done because you have no other way to get answers to your questions. If you already know the answer, there is no reason for the expense of a prototyping effort. I know of one project that ignored an actual problem (can we really get these several systems to share data) and prototyped something they already knew the answer to (what is a third party's process for doing inventory). Needless to say, the unanswered question came up later, and quite a bit of investigation still had to be done to find a solution. They wasted a lot of time prototyping something that was completely known, and still had the later expense of prototyping and testing solutions for the unanswered question.

Slide 9



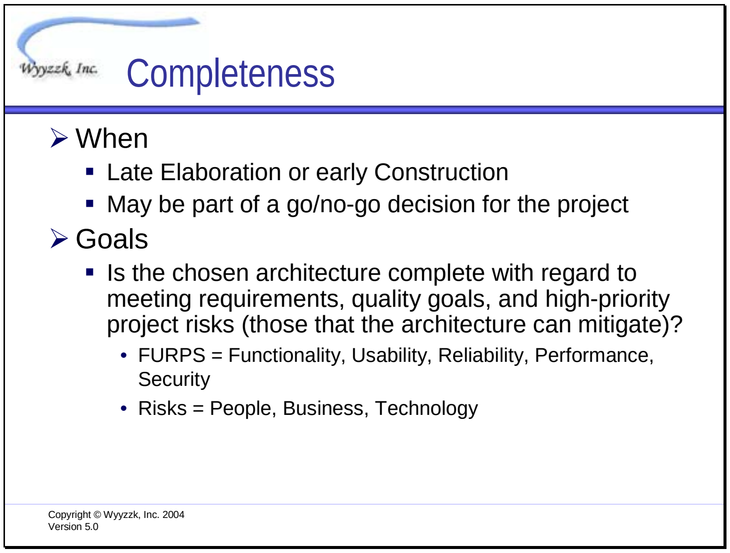
Early Feasibility

- When
  - Typically Late Inception or Early Elaboration
  - May be part of a go/no-go decision for the project
  - May be part of a suitability review
- Goals
  - Is this a feasible architecture for this project team or this company?
    - Cost of development and supporting technologies
    - Time to develop
    - Team skills – do we have the skills we need or can quickly acquire them
    - Maturity of technology – are we risking our project on version 1 of a technology?

In this review, we are assuming that the architecture is suitable for the project. Now we look further to determine if we could really create a project using this architecture, given the people on the team, their skill level, and the availability of supporting technologies (Eclipse, .Net, C#, Java, etc.)

If we do not have the appropriate skills or technologies, what would it cost to acquire the technologies, install the technologies, train the project team and support staff, and hire an expert on the technologies to mentor the team? Can we afford that in time or money?

Slide 10



Wyyzzk, Inc.

Completeness

- When
  - Late Elaboration or early Construction
  - May be part of a go/no-go decision for the project
- Goals
  - Is the chosen architecture complete with regard to meeting requirements, quality goals, and high-priority project risks (those that the architecture can mitigate)?
    - FURPS = Functionality, Usability, Reliability, Performance, Security
    - Risks = People, Business, Technology

Version 5.0

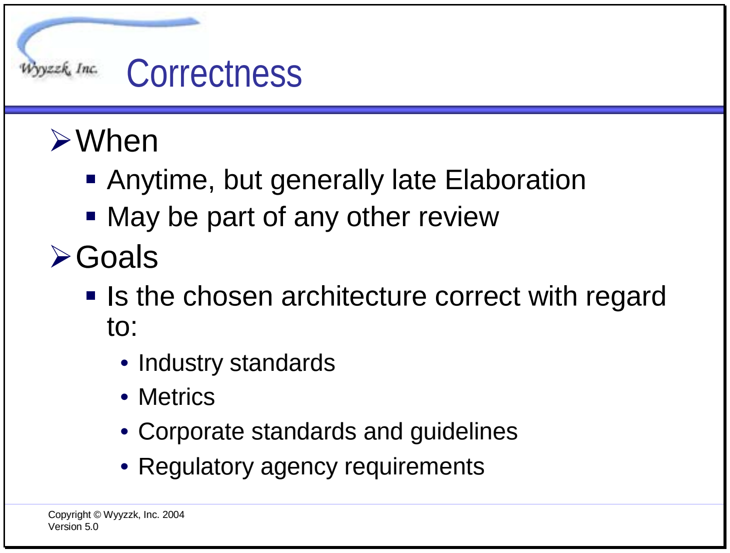
Risks are things that can cause your project to fail. There are several categories of risk:

People – can we put the right team together, do they need training, do we need to hire outside resources, what if the team does not get along, what if we cannot get a vital person onto the team, what if someone really vital to the project has to leave the project for any reason

Business – what if a change happens in the marketplace – can we respond, what if we are too late or too early to market, will the company exist long enough to get the project done, do we have the time and money we need to complete the project

Technology – is the technology we are using known or unknown, how many companies support this technology, what if the technology does not meet our needs after we have committed to it, can we afford the cost of licenses, installation, and training (in either time or money)

Slide 11

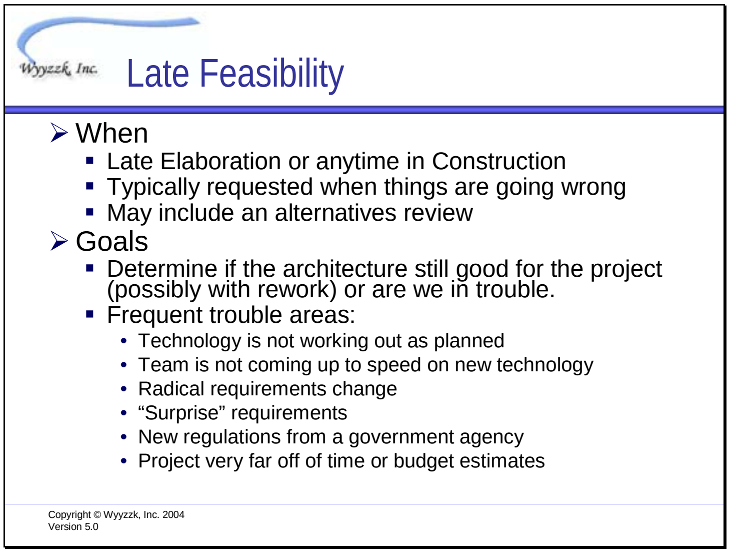


Correctness

- When
  - Anytime, but generally late Elaboration
  - May be part of any other review
- Goals
  - Is the chosen architecture correct with regard to:
    - Industry standards
    - Metrics
    - Corporate standards and guidelines
    - Regulatory agency requirements

In addition to the notes in the Testing Architecture section, you may also need to review the architecture for adherence to corporate standards or requirements of regulatory agencies (such as Federal Aviation Administration (FAA) for flight control software or air traffic control systems, Food and Drug Administration (FDA) for grocery stores, pharmacies, research facilities, or medical instrumentation, HIPPA requirements for handling personal information, and Sarbanes-Oxley for reporting on corporate processes).

Slide 12



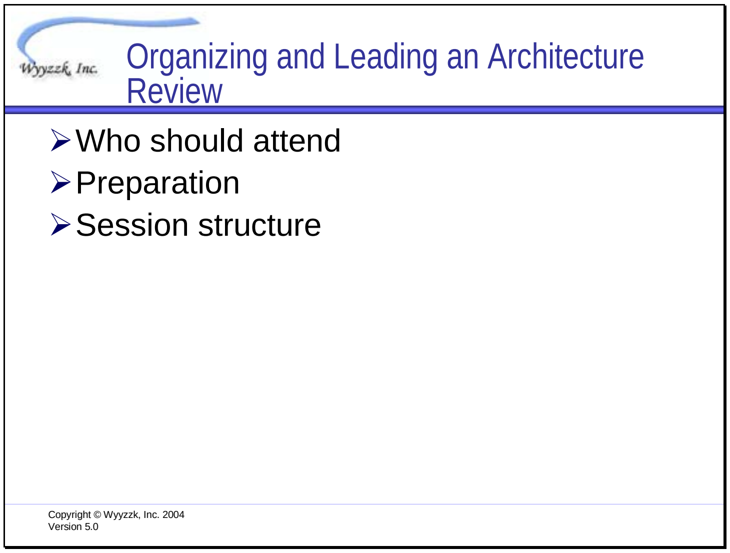
Late Feasibility

- When
  - Late Elaboration or anytime in Construction
  - Typically requested when things are going wrong
  - May include an alternatives review
- Goals
  - Determine if the architecture still good for the project (possibly with rework) or are we in trouble.
  - Frequent trouble areas:
    - Technology is not working out as planned
    - Team is not coming up to speed on new technology
    - Radical requirements change
    - “Surprise” requirements
    - New regulations from a government agency
    - Project very far off of time or budget estimates

Version 5.0

In this situation, generally the company thinks they can still recover the project. Often this kind of review is to determine what is not working and identify what needs to be fixed. The outcome may be that the architecture (and the project) can be fixed, or that the project should just be cancelled at this point.

Slide 13

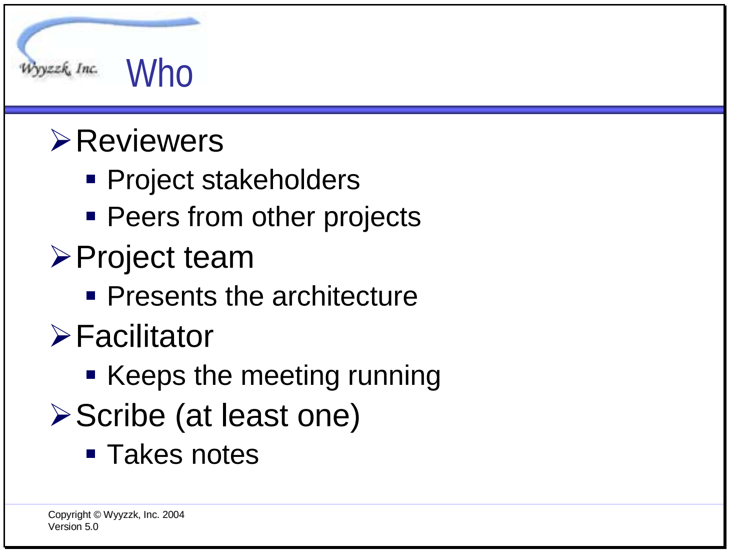


Organizing and Leading an Architecture Review

- Who should attend
- Preparation
- Session structure

Version 5.0

Slide 14



Who

- ➢ Reviewers
  - Project stakeholders
  - Peers from other projects
- ➢ Project team
  - Presents the architecture
- ➢ Facilitator
  - Keeps the meeting running
- ➢ Scribe (at least one)
  - Takes notes

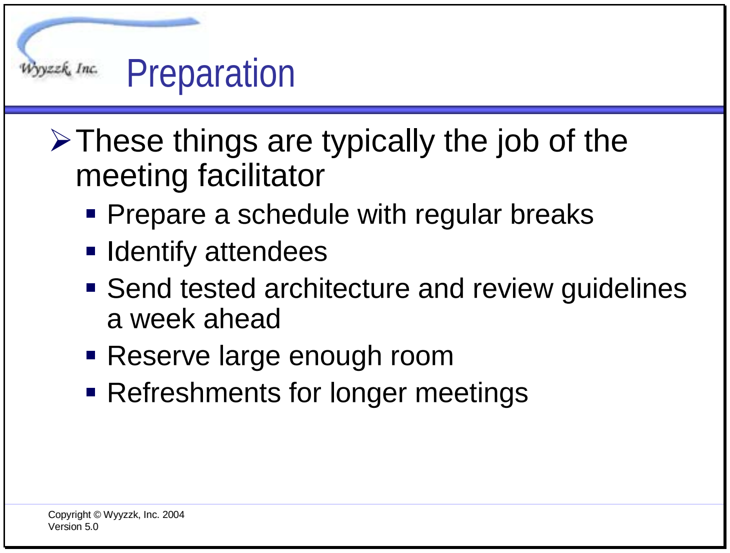
Project stakeholders can include users, people representing the business needs of the company, and the project sponsor. It may also include senior designer/developers from the project team.
Peers from other projects – usually other architects with experience in the company or with this type of project or both.

Project team – typically the architecture team and the project manager. Possibly also a senior designer.

Facilitator – not a member of the presentation group. Someone whose job it is to run the meeting and keep it on track. This could be the project manager or BA if that person is not in another role. It can even be someone outside the project team – some companies have people whose job is to facilitate meetings. Sometimes this is an outside vendor who is running the architecture review (such as SEI for example).

Scribe – not a member of the presentation group. Someone whose job it is to take notes. Often the Business Analyst for the project takes on this role. In large projects, you may need more than one scribe to keep up with the meeting. Sometimes we have one making notes on a whiteboard or easel pad, while another is taking notes in Word or on a paper tablet.

Slide 15



Wyyzzk, Inc.

Preparation

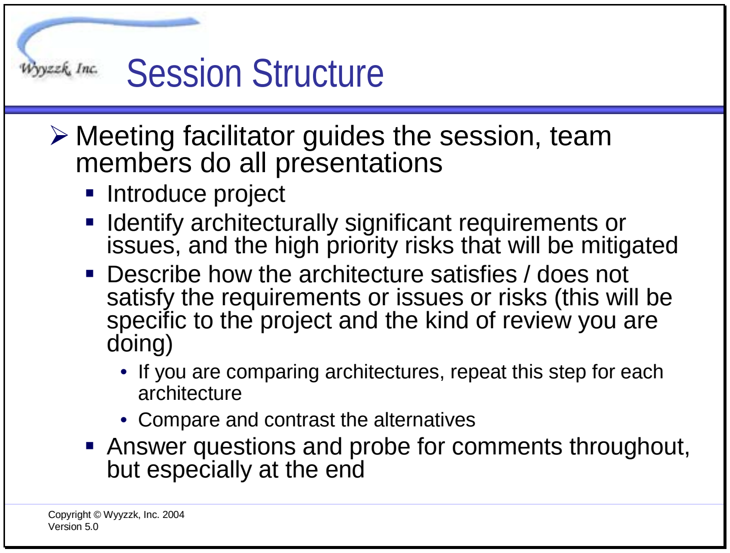
- These things are typically the job of the meeting facilitator
  - Prepare a schedule with regular breaks
  - Identify attendees
  - Send tested architecture and review guidelines a week ahead
  - Reserve large enough room
  - Refreshments for longer meetings

Length of review could be anywhere from a couple of hours to several days, depending on the size of the project

For long review sessions, it is possible not everyone needs to be at the whole review. Let people know what parts they are expected to attend and when those parts are scheduled

Note architecture should already be tested as much as possible based on the current completeness of the architecture (using ideas from testing notes). Architecture is not necessarily complete when reviewed. Review guidelines will include architecturally significant requirements and issues, as well as the type of architectural review you will be holding

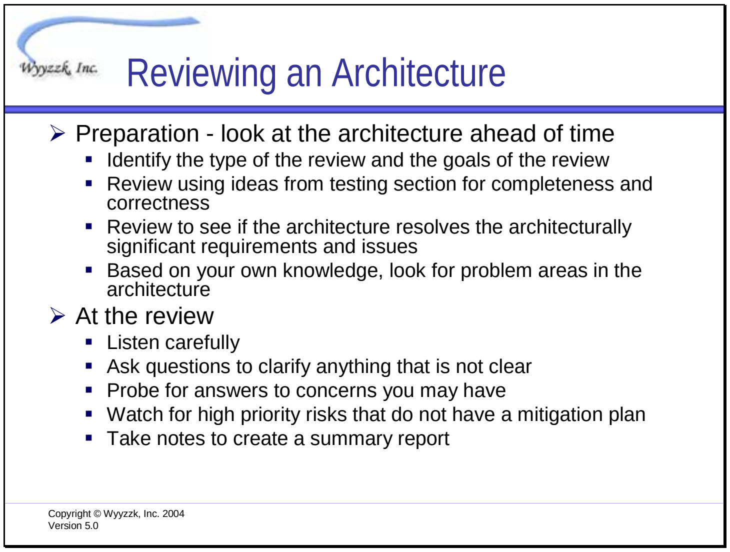
Slide 16



Session Structure

- Meeting facilitator guides the session, team members do all presentations
  - Introduce project
  - Identify architecturally significant requirements or issues, and the high priority risks that will be mitigated
  - Describe how the architecture satisfies / does not satisfy the requirements or issues or risks (this will be specific to the project and the kind of review you are doing)
    - If you are comparing architectures, repeat this step for each architecture
    - Compare and contrast the alternatives
  - Answer questions and probe for comments throughout, but especially at the end

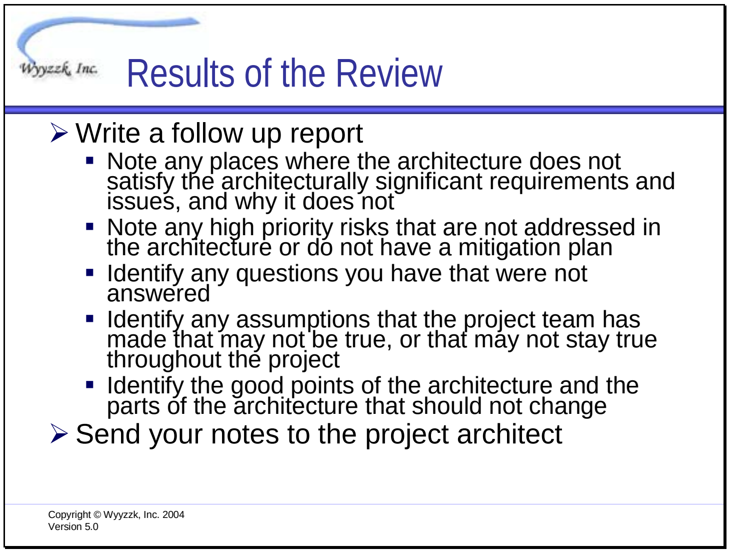
Slide 17



Reviewing an Architecture

- Preparation - look at the architecture ahead of time
  - Identify the type of the review and the goals of the review
  - Review using ideas from testing section for completeness and correctness
  - Review to see if the architecture resolves the architecturally significant requirements and issues
  - Based on your own knowledge, look for problem areas in the architecture
- At the review
  - Listen carefully
  - Ask questions to clarify anything that is not clear
  - Probe for answers to concerns you may have
  - Watch for high priority risks that do not have a mitigation plan
  - Take notes to create a summary report

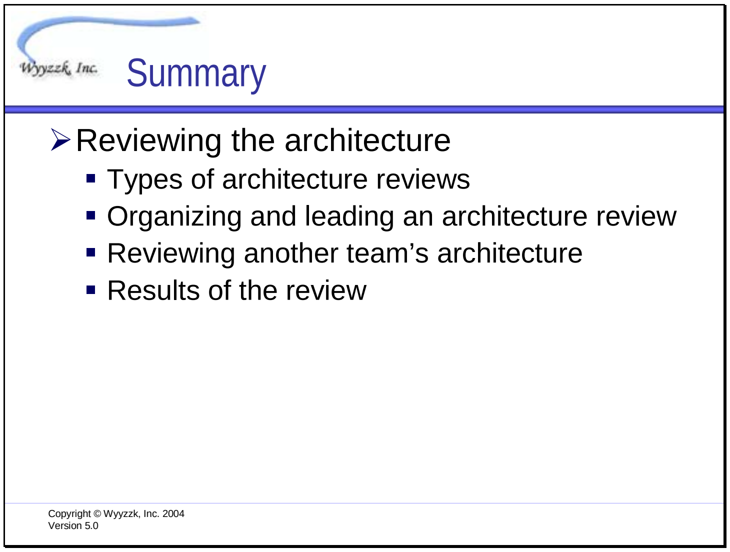
Slide 18



Results of the Review

- Write a follow up report
  - Note any places where the architecture does not satisfy the architecturally significant requirements and issues, and why it does not
  - Note any high priority risks that are not addressed in the architecture or do not have a mitigation plan
  - Identify any questions you have that were not answered
  - Identify any assumptions that the project team has made that may not be true, or that may not stay true throughout the project
  - Identify the good points of the architecture and the parts of the architecture that should not change
- Send your notes to the project architect

Slide 19



Summary

- Reviewing the architecture
  - Types of architecture reviews
  - Organizing and leading an architecture review
  - Reviewing another team’s architecture
  - Results of the review