Future Directions in Product Testing

Vikram G Shah

Network Appliance

Why is this topic of Interest to us?

- **The automobile industry spends $2 billion to $3 billion per year fixing software problems**
- **Software bug was a major contributor to the famous 2003 Northeast blackout in US – economic losses estimated at $6 billion**
- **In March 2002 software bugs in Britain's national tax system resulted in more than 100,000 erroneous tax overcharges**
- **Major European railroad hit by the aftereffects of the Y2K bug – many trains would not run due to inability to recognize the date 31/12/2000**
- **Cost of finding and fixing defects is 25% of overall India's GDP for IT sector**

Why is this topic of Interest to us?

- Data presented represents only the tip of the iceberg…. **losses due to poor quality software** runs into tens of billions of dollars every year
- As software becomes more complex, schedules become shorter due to market demands and competitive forces, **poor quality will continue to be a huge challenge** in the years to come
- Poor / inadequate testing **only one of the major reasons for poor quality**….issues with understanding of requirements and lack of early focus on quality equally major issues

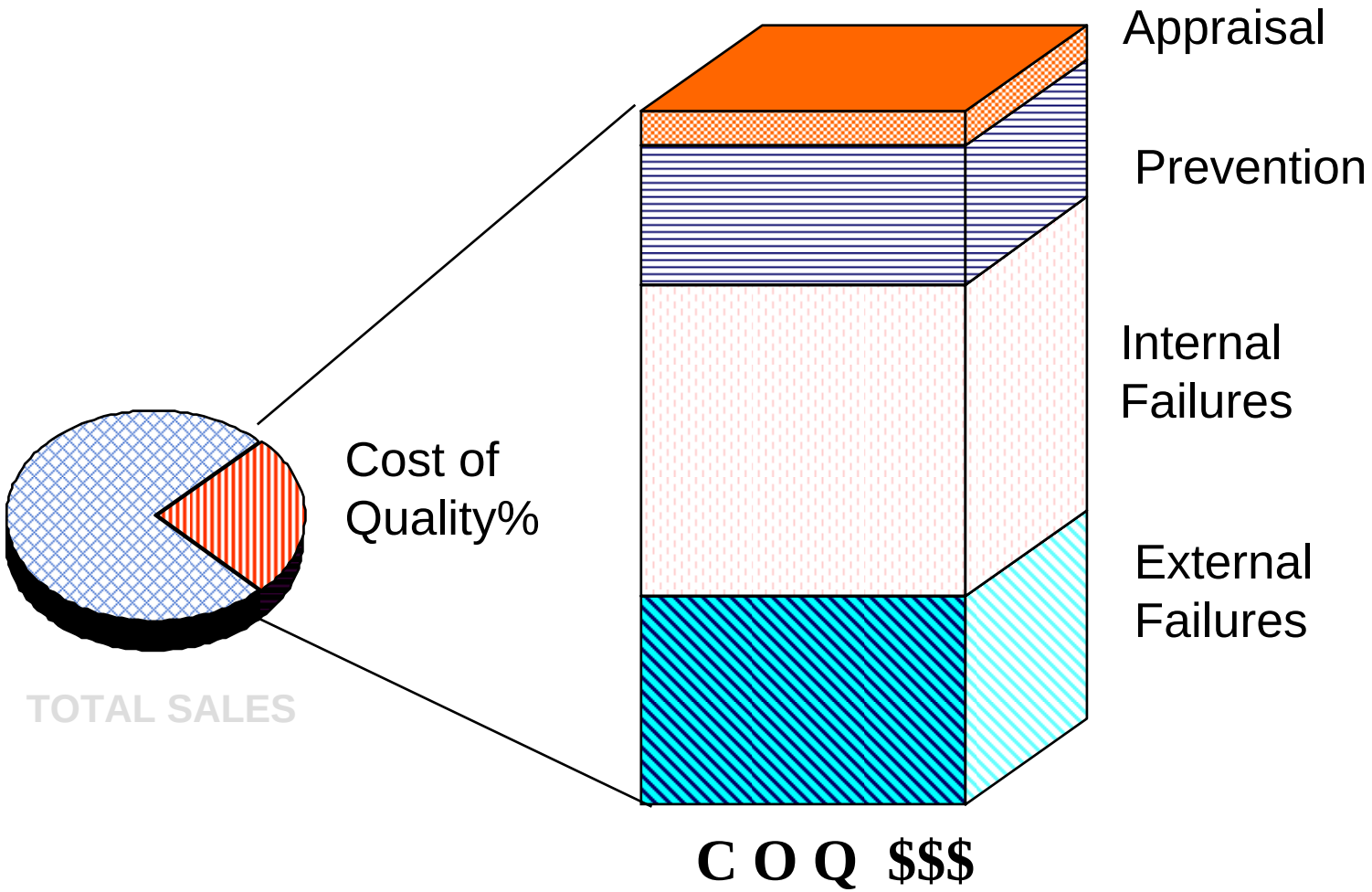
In short, strategic importance of QA and testing does not need to be overstated

How must testing is enough?

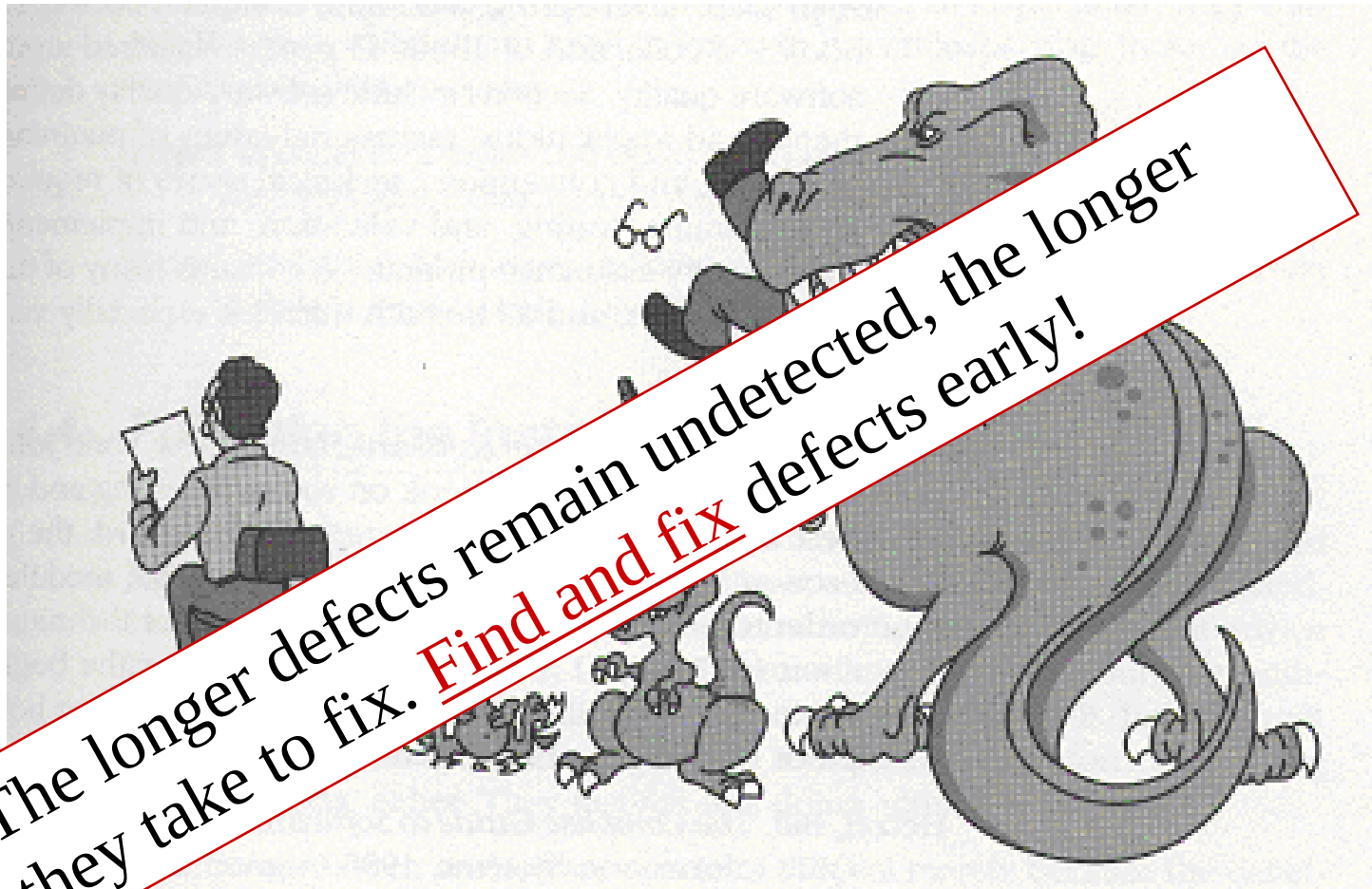
If 99.9% were Good Enough

- 9,703 checks would be deducted from the wrong bank accounts each hour
- 27,800 pieces of mail would be lost per hour
- 3,000,000 incorrect drug prescriptions per year
- 8,605 commercial aircraft takeoffs would annually result in crashes

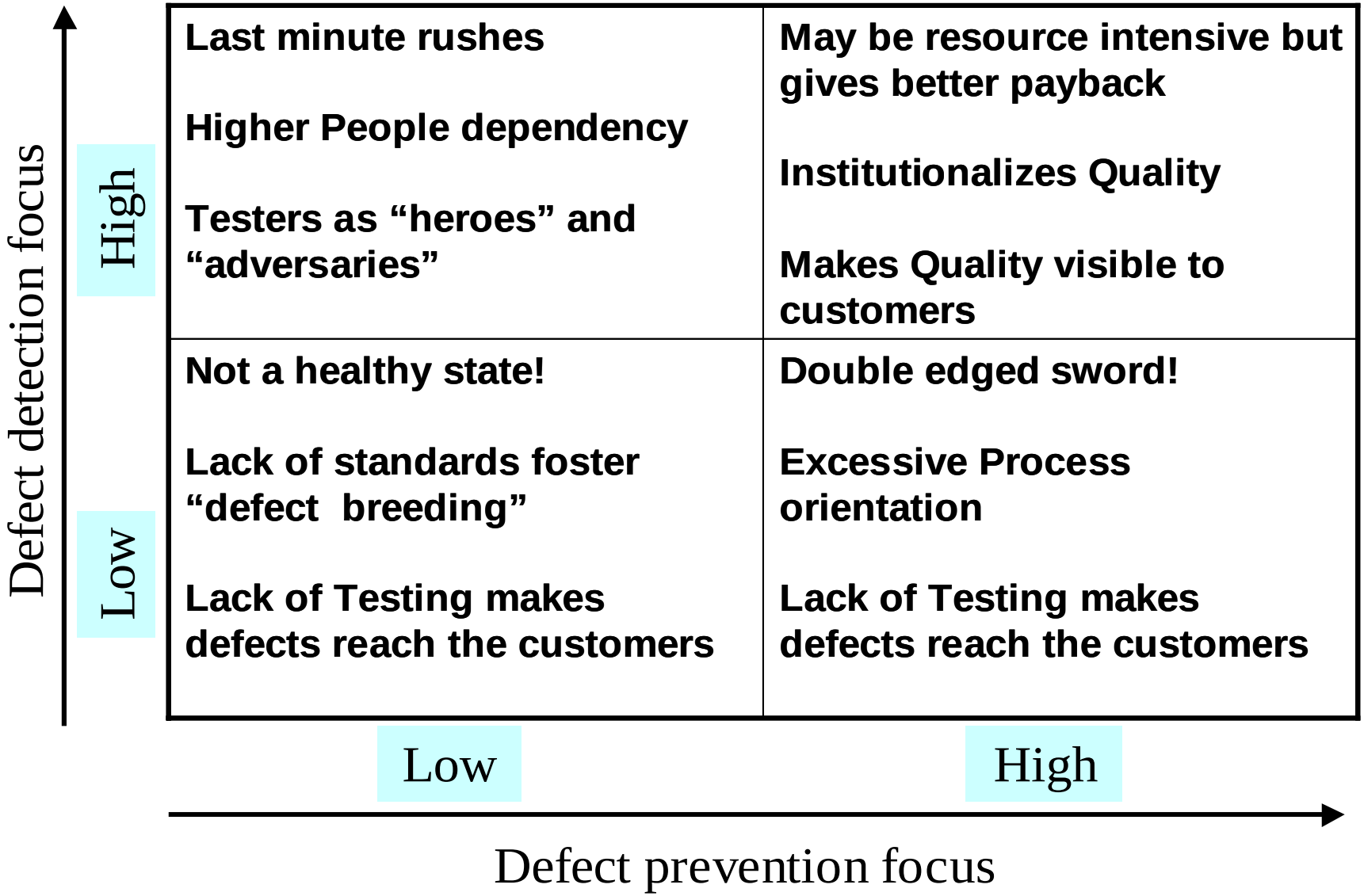
Cost of Quality



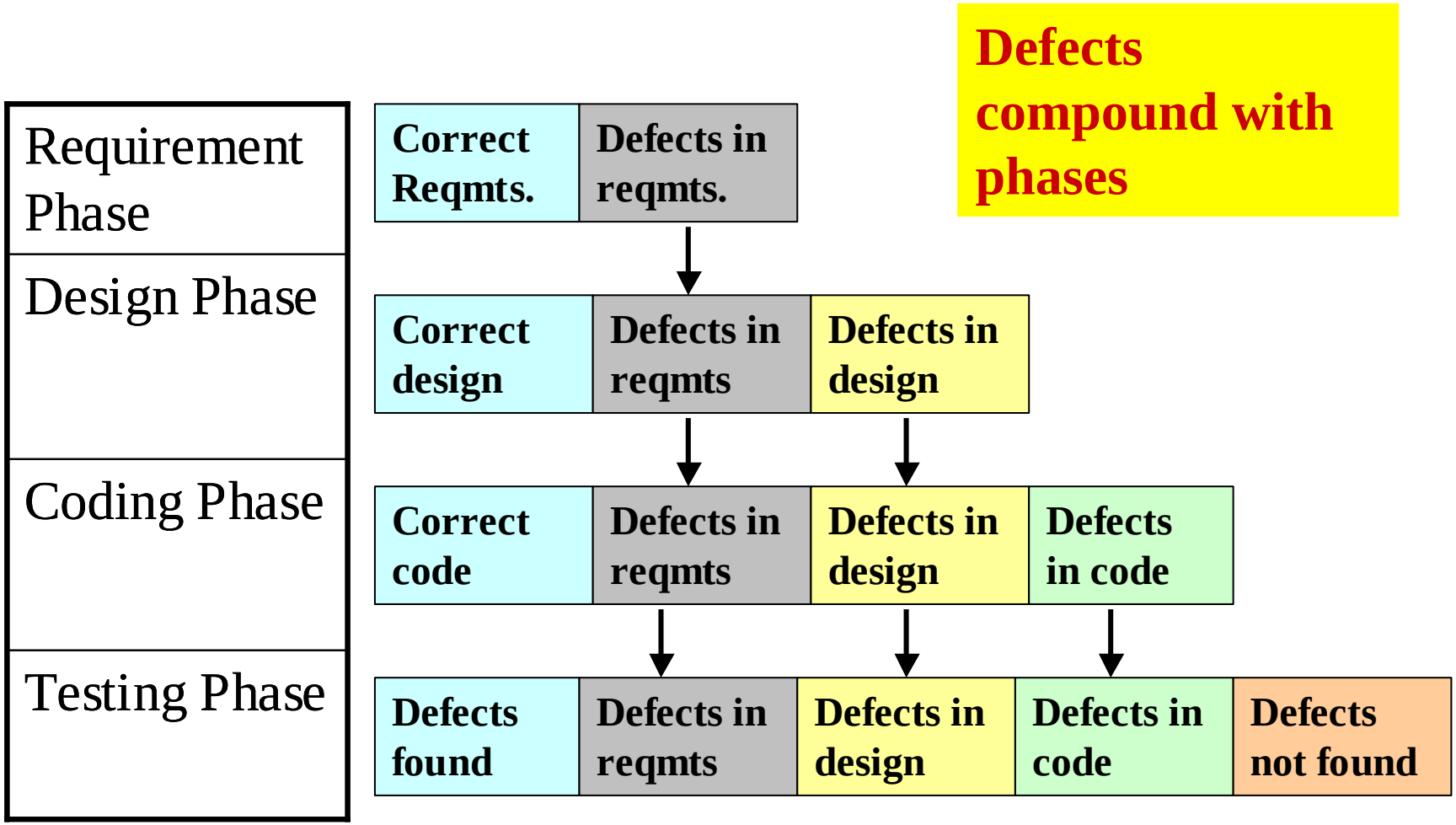
Fix defects before they become monsters !



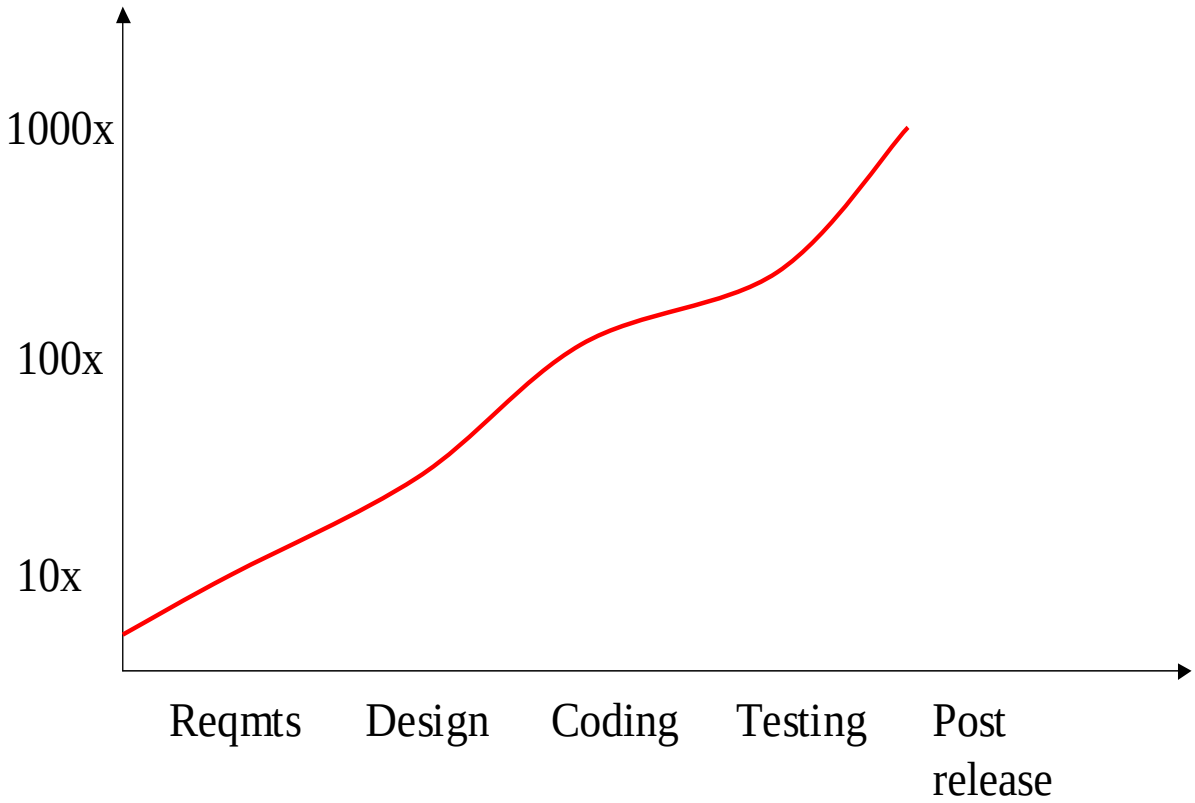
Defect Prevention and Detection



Compounding Effect



Compounding Cost of fixing defects



Defects should be detected and corrected at the earliest – Postponing detection of defects compounds cost!

Experience Sharing—NetApp Way

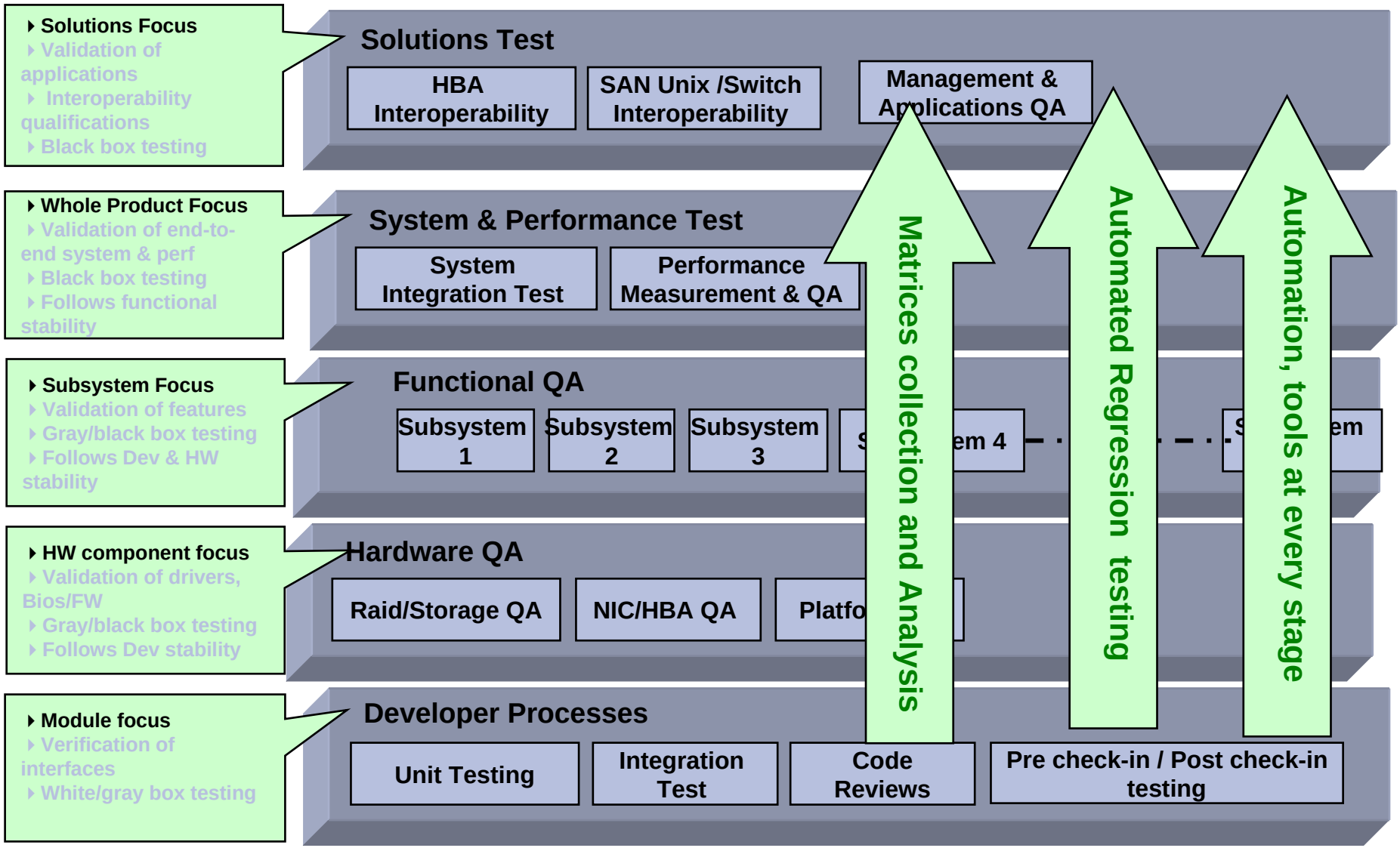
1996	2007
Engineers: <100 **Customers:** Yahoo, Tandem, Synopsys… *"Early adopters, conservative"*	**Engineers:** >1500 **Customers:** Airbus, Edward Jones, Charles Schwab, *"Enterprise"*
Customers' priorities: - High performance - Latest technology - Advanced features - Responsive vendor	**Customers' priorities:** - Stability - Excellent reliability - High performance - Proactive vendor
Engineering priorities: - Fast - Simple - Reliable	**Engineering priorities:** - Reliable - Fast - Simple

Experience Sharing—NetApp Way

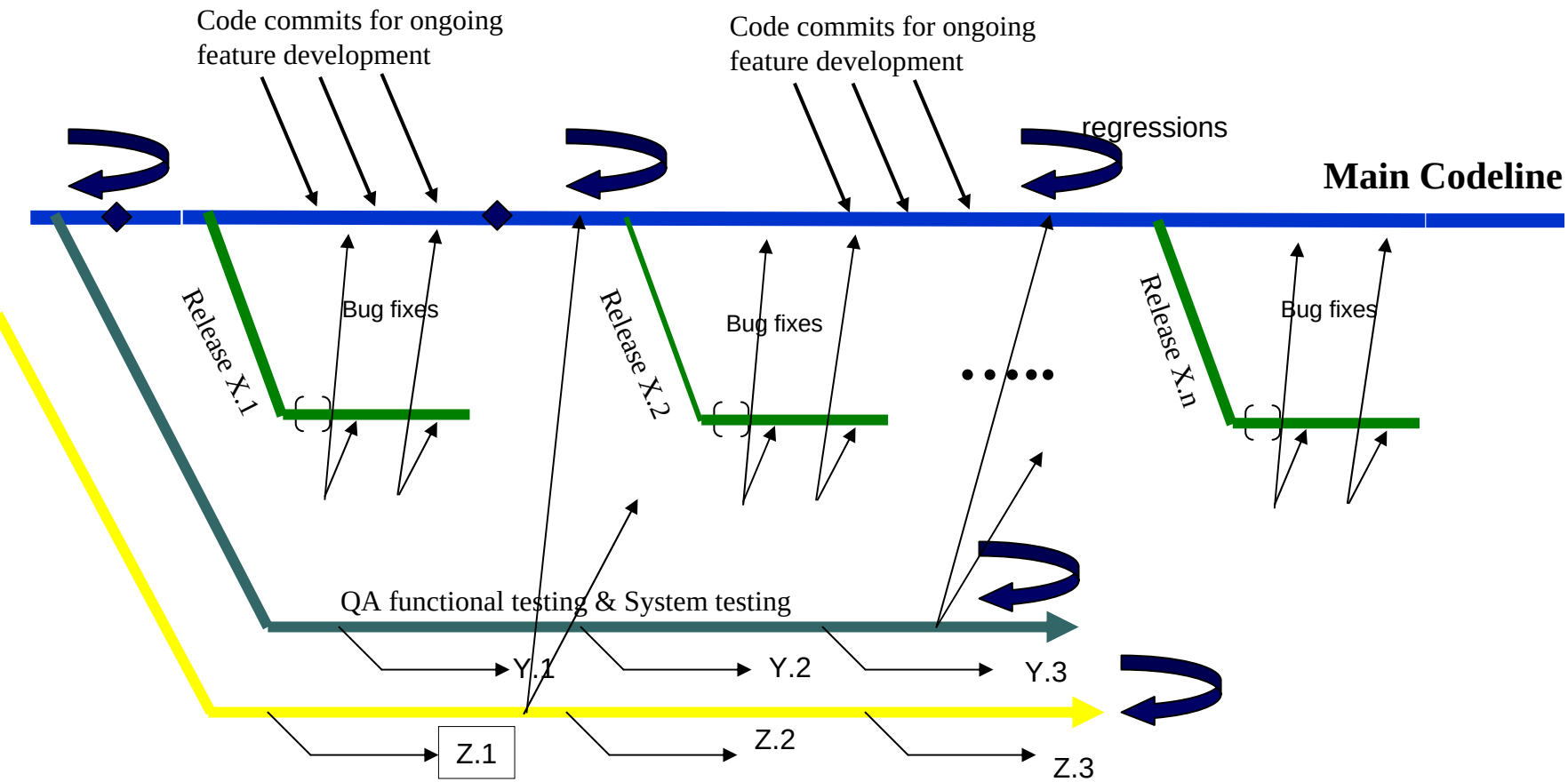
1996	2007
Keys to success: - Bright & dedicated people - Small teams - Rapid prototyping	**Keys to success:** - Bright & dedicated people - Development life cycle with well-defined processes - Extensive metrics
Quality assurance model: - Back-end testing	**Quality assuran** - Planning & - Inspe - R esting pired Testing s improvement

Software Processes are evolving ……

QA Coverage---Layered Approach



Product Testing Challenge---Multiple releases



- **Commits to Main codeline for new feature development occurs on continuous basis**
- **Ongoing system test and Regression occurs on Main codeline to keep sanity on Main**
- **Functional, system test and Regression happens on each release branch on an ongoing basis**
- **Testing on sustaining releases is additional effort**

Key factors to reduce COQ

COQ = Cost of Prevention+ Cost of Detection + Cost of Correction

- Product Organizations MUST cut down on **regression costs** – this is a recurring Cost. (varies with release models and can be a trap if not addressed)

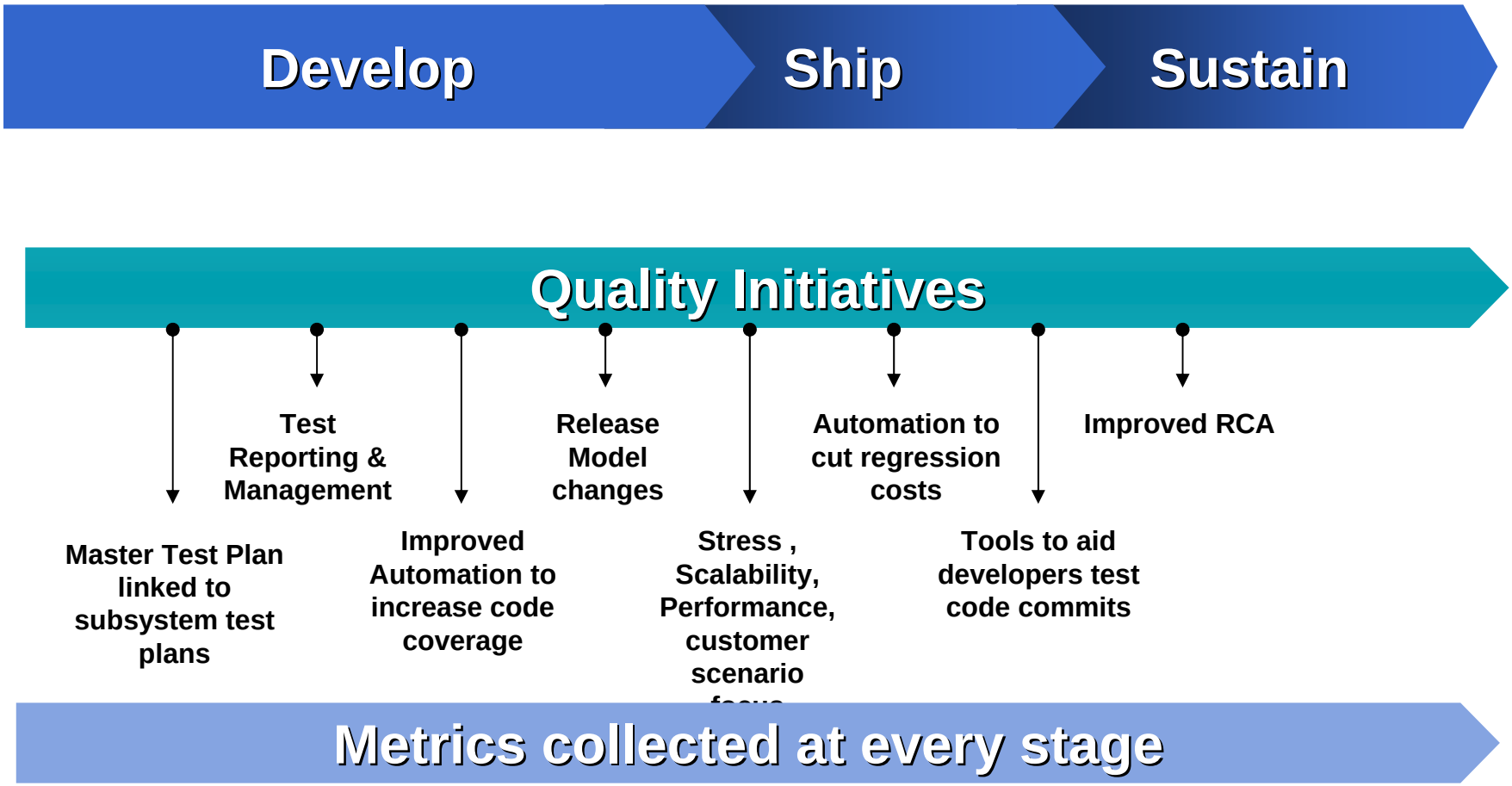
- Focus on **Test Infrastructure** to enable robust **test automation** and execution.

- Focus on **Code Coverage** through automated tests to increase product stability and reduce regressions.

- Focus on shared **Test hardware** and **Test environment.**

- Focus on Tools to increase **developer and test productivity** and leverage common set of tools.

Quality Improvement Initiatives in a Product organization



Future Directions in Software Product Testing

Objective	Future Directions
Cut down on Cost of Quality, Regression costs	▸ Test driven development ▸ Improve automation coverage to close to 100% ▸ Prevent more defects and detect less defects by learning from other industries (eg. Automobile industry)
Reduce external failures, Improve reliability & performance from customer point of view	▸ Follow Agile methodologies, extreme programming, SCRUM ▸ Increase QA coverage by involving multiple parties (Buddy testing, customer summits, Beta …etc). "Testing isn't for test teams alone anymore" ▸ Collaborative computing: Prevent defects by involving test engineers at coding, Design "Development isn't for developers alone anymore" ▸ Periodic reviews with customers and customer visits by testing team
Reduce release cycle time	▸ Parallel releases ▸ Effective regression testing ▸ Eliminate wastage (muda) through better processes, communication ▸ Have people who can cross boundaries with different roles such as Developer & tester (All-rounders)
Improve return on investment from testing activities	▸ "More with Less" by improving automation & manual testing productivity ▸ Effective metrics that relates test productivity and company costs

Thank You

(vshah@netapp.com)