

NAISTO EASE PROJECT

NASSCOM Quality Summit

The Software Industry in Japan and Empirical Software Engineering

Koji Torii
Executive Director, Professor
EASE Project, Nara Institute of Science and Technology


Today's agenda

- Japan's situation with regard to software industries, i.e.,
 - ◆ Examples of recent aspects/accidents/topics in Japan i.e., crisis, measurement-base, embedded
- Japan SEC (Software Engineering Center) under METI
- Empirical Software Engineering
- EASE Project under MEXT
 - ◆ The introduction of empirical software engineering
- EPM (Empirical Project Monitor) in EASE
 - ◆ Automatic data collection system
 - ◆ Open Source System
- Data Analysis with EPM
- Summary

NASSCOM Quality Summit NAISTO EASE PROJECT

Making software reliable and improving productivity by using measurement data

- In the car industry over one million improvements per year are made based on measurement data obtained from development sites, and a saving of \$1 billion US a year is achieved.
- Point of sales systems
 - ◆ Systems that record sales information about commodities sold in supermarkets and local convenience stores then use the result to **control stock and for marketing purposes**.
 - ◆ The sales trends of two or more stores can be compared, and the data can be **linked to other data including weather data**, which allows overlaps and tendencies to be understood.



NASSCOM Quality Summit NAISTO EASE PROJECT

Smart counting ~ sushi-conveyor-belt case

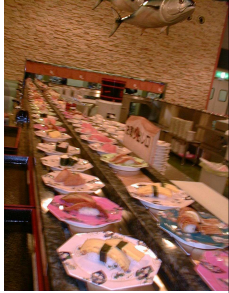
Traditionally

- ◆ Diners pay for food based on the number of plates remaining.
- ◆ Waiters counted the number of plates (price per plate differs according to colour; e.g.: Red = \$1, Green = \$2, Blue = \$3).

↓

Problems with this traditional counting method:

- ◆ Time consuming
- ◆ Prone to mistakes



NASSCOM Quality Summit NAISTO EASE PROJECT

Smart counting ~ sushi-conveyor-belt case (cont.)

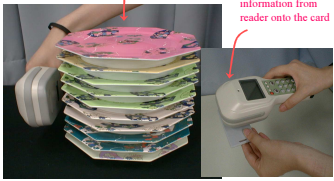
Breakthrough: RFID in Use

RFID tags embedded in sushi plates

① Automatic Count
(Time efficient, Accurate count)


Sushi plates with RFID tags

Transmitting information from reader onto the card



② Automatic reading at the checkout counter
(Time efficient, Accurate count)

Automatic payoff at the checkout



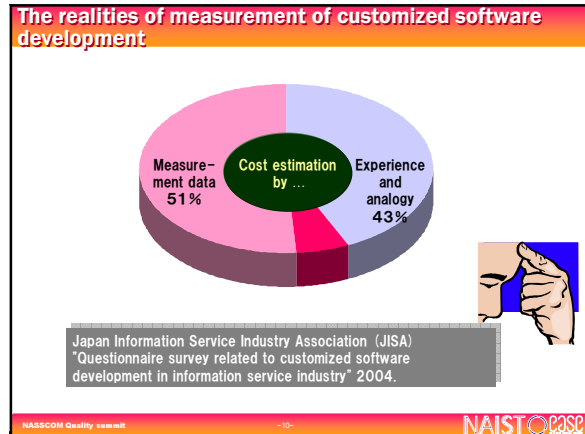
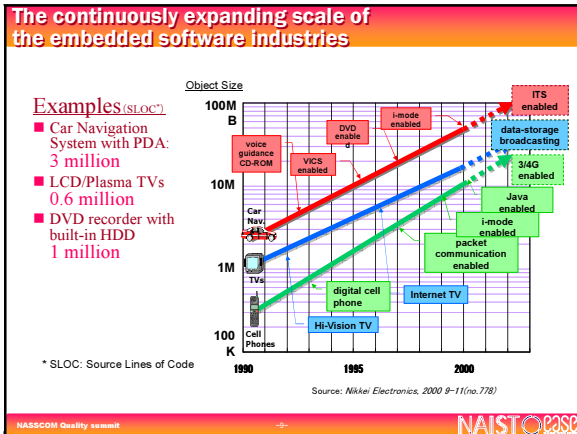
NASSCOM Quality Summit NAISTO EASE PROJECT

Profile of the Embedded Software Industry in Japan

- The Embedded Software Industry
 - ◆ Engineers: 150,000.
 - ◆ Development size: \$20 billion US.
 - Output of embedded systems (dynamic statistics): \$500 billion US.
 - ◆ Output: about 10 billion lines/year
 - One engineer develops 6,000 lines/year on average.
- Comparison Figures from the Information Service Industry
 - ◆ Employees: 510,000
 - ◆ Sales: \$140 billion US

Source: Report on Embedded Software Industries (The Ministry of Economy, Trade and Industries, 2004)

NASSCOM Quality Summit NAISTO EASE PROJECT



Japan SEC
(Japan Software Engineering Center)

SEC

The Japan Software Engineering Center

- Opened in October 2004 supported by the Ministry of Economy, Trade and Industry (METI).
- Budget of 2004: \$14.8 million US

■ The SEC mission

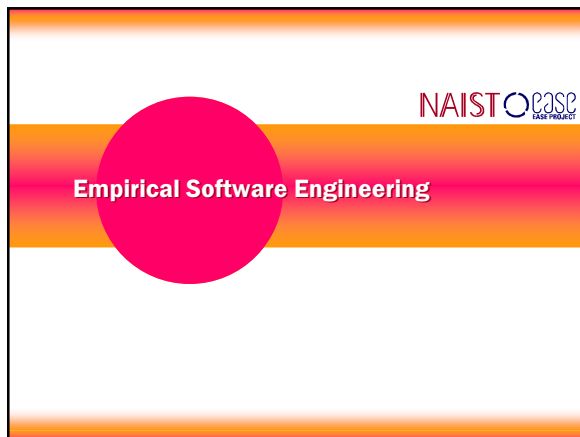
- ◆ To provide **common understanding** of
 - Data-oriented quantitative approaches
 - Collect real **project data (quantitative and qualitative)**
 - Create **benchmarks**
 - Best practices
 - Provide guidelines about **roles and recommended activities**.

The SEC activities and approaches

- Conduct in-depth practical studies to solve the issues of today's software industry.
 - ◆ Software Process Improvement methods for the Japanese Industry
 - ◆ Software measurement standards
 - Gather data from various software development projects underway.
 - Analyze these quantitative data.
 - Promote the use of **measurement standards**.
 - ◆ Demonstration of the methods and tools in advanced software development projects

The SEC projects

- Software engineering for enterprise systems
 - ◆ Improve quality and productivity
- Software engineering for embedded technology
 - ◆ Strengthen development capability,
 - ◆ Nurture skilled engineers,
- Advanced software development
 - ◆ Building best practices



Why empirical software engineering? (Basili)

- **Understanding a discipline** involves building models, e.g., application domain, problem solving processes
- **Checking** our understanding is correct, e.g., testing our models, experimenting in the real world
- **Analyzing** the results e.g. learning, the encapsulation of knowledge
- This is the **empirical paradigm** that has been used in many fields, e.g., physics, medicine, manufacturing
- Like **other disciplines**, software engineering requires an empirical paradigm

NASSCOM Quality summit -16- NAIST Obase

Why empirical software engineering? (Basili)

- Empirical software engineering requires
 - ◆ the scientific use of **quantitative and qualitative data**
 - ◆ about the software **product**, software development **process** and software **management**
- It **requires real world laboratories**
 - ◆ Research needs laboratories to observe and manipulate the variables
 - ◆ Development needs models to understand how to build systems better
- **Research and Development** have a symbiotic relationship

NASSCOM Quality summit -17- NAIST Obase

Observations, laws and theories (Rombach)

- **Empirical observations**
 - ◆ Facts from individual empirical studies.
 - ◆ We can **characterize** phenomena based on them.
- **Laws**
 - ◆ Repeatable observations.
 - ◆ We can understand context enough to make prediction about future observations.
 - ◆ We can **predict** phenomena by them. (what)
- **Theories**
 - ◆ Cause-effect relationships.
 - ◆ We can **explain** phenomena by them. (why)

A. Endres and D. Rombach, (2003). A Handbook of Software and Systems Engineering: Empirical Observations, Laws and Theories. Essex, England:

NASSCOM Quality summit -18- NAIST Obase

Journal by Kluwer

Empirical Software Engineering

- **Scope**
 - ◆ Cost estimation techniques
 - ◆ Analysis of the effects of design methods and characteristics
 - ◆ Evaluation of testing methodologies
 - ◆ Development of predictive models of defect rates and reliability from real data
 - ◆ Infrastructure issues, such as measurement theory, experimental design, qualitative modeling and analysis approaches.

...

NASSCOM Quality summit -19- NAIST Obase

ISESE

International Symposium on Empirical Software Engineering

ISESE2002 @ Nara, Japan
 ISESE2003 @ Roma, Italy
 ISESE2004 @ California, USA
 ISESE2005 @ Noosa Heads, AU

Nov.17-18
<http://attend.it.uts.edu.au/ise2005/>

NASSCOM Quality summit -20- NAIST Obase

ISERN
International Software Engineering Research Network

- ISERN is a community that believes software engineering research needs to be performed in an experimental context.
- ISERN was established in 1993 by researchers of software engineering from 12 countries, including USA, Germany, Australia, Italy, Finland and Japan.
- ISERN provides several means of communication between members;
 - ◆ Electronic Communication,
 - ◆ Annual meetings, and
 - ◆ Exchange of researchers.

NASSCOM Quality Summit -21- NAIST EASE

NAIST EASE

The EASE Project

NASSCOM Quality Summit -22- NAIST EASE

What is the EASE project?

- Empirical Approach to Software Engineering
- One of the leading projects of the Ministry of Education, Culture, Sports, Science and Technology (MEXT).
- 5 year project starting in 2003.
- Budget: \$2 million US / year.
- Project leader: Koji Torii, NAIST
 - Sub-leader: Katsuro Inoue, Osaka University
 - Kenichi Matsumoto, NAIST

<http://www.empirical.jp/English/>

NAIST EASE Project
Empirical Approach to Software Engineering

NASSCOM Quality Summit -23- NAIST EASE

The purpose of the EASE project

- Achievement of software development technology based on quantitative data
 - ◆ Construction of a quantitative data collection system
 - Result 1: Making of EPM open source
 - ◆ Construction of a system that supports development based on analyzed data
 - Result 2: EPM application experience
 - Result 3: Coordinated cooperation with SEC
- Spread and promotion of software development technology based on quantitative data to industry sites
 - Result 4: Activation of the industrial world

NASSCOM Quality Summit -24- NAIST EASE

Empirical activities in EASE

- Data collection in real time, e.g.
 - ◆ configuration management history
 - ◆ issue tracking history
 - ◆ e-mail communication history
- Analysis with software tools, e.g.
 - ◆ metrics measurement
 - ◆ project categorization
 - ◆ collaborative filtering
 - ◆ software component retrieval
- Feedback to stakeholders for improvement, e.g.
 - ◆ observations and rules
 - ◆ experiences and instances in previous projects

NASSCOM Quality Summit -25- NAIST EASE

The EASE roadmap

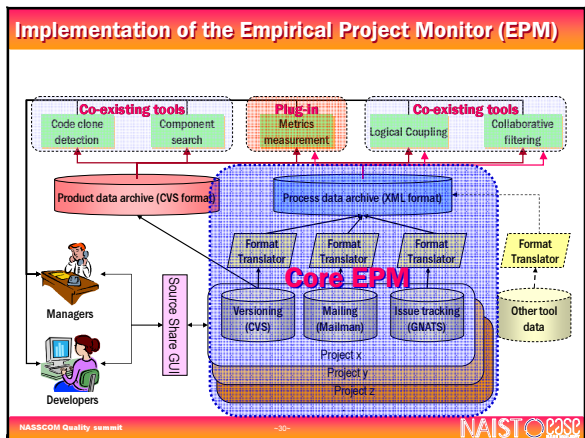
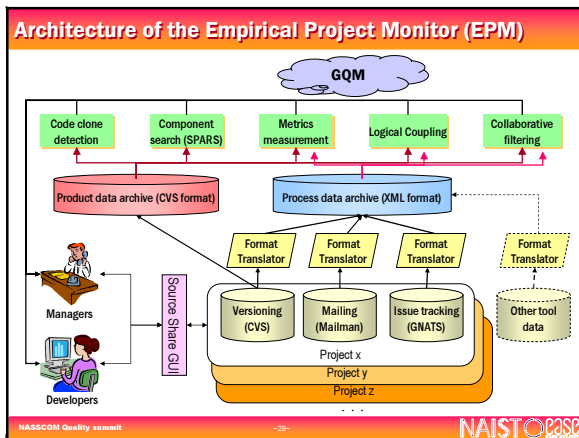
NASSCOM Quality Summit -26- NAIST EASE



EPM
The Empirical Project Monitor

- An application supporting empirical software engineering
- EPM automatically collects development data accumulated in development tools through everyday development activities
 - ◆ Configuration management system: CVS
 - ◆ Issue tracking systems: GNATS
 - ◆ Mailing list managers: Mailman, Majordomo, FML

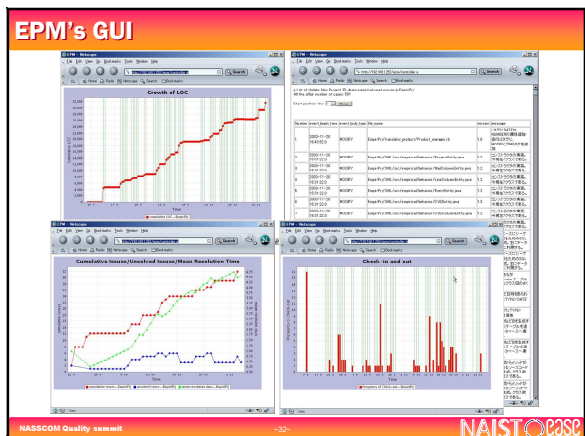
NASSCOM Quality Summit -29- NAIST CASE



Automated data collection in EPM

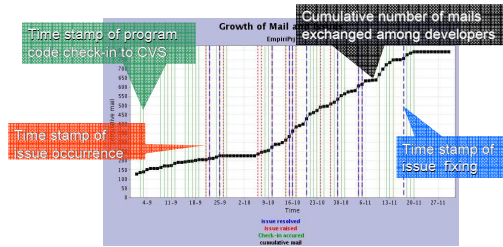
- Reduces the reporting burden on developers
 - ◆ without additional work for developers
- Reduces the project information delay
 - ◆ data available in real time
- Avoids mistakes and estimation errors
 - ◆ uses real (quantitative) data

NASSCOM Quality Summit -31- NAIST CASE



An example of output

- EPM can put data collected by CVS, Mailman, and GNATS together into one graph.



Merits to introducing EPM

- Easy monitoring of projects in cooperation with the existing development environment.
- Easy accumulation of the knowledge and experience of projects.
- Collection and sharing of uniform data for projects in real time.
- Sharing and reuse of information enabled through empirical data.

Data Analysis with EPM

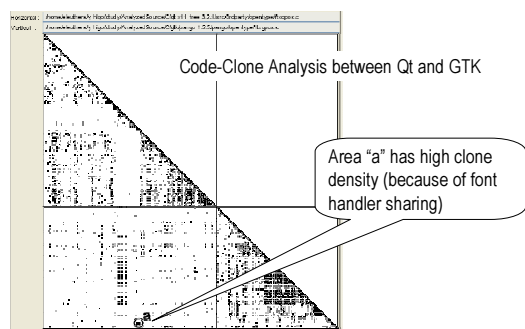
How can we use so much data?

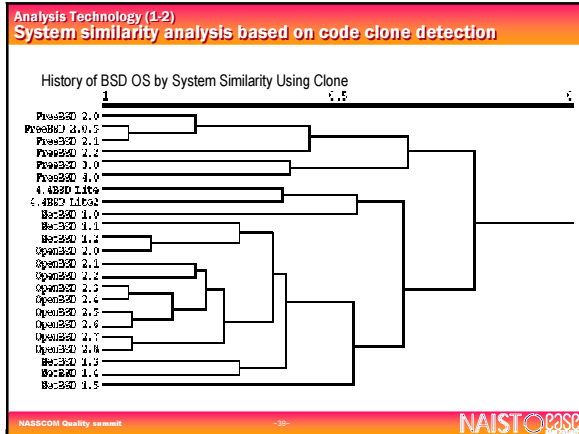


Analysis technologies and models

- Code clone analysis (CCfinder)
- Collection, analysis, and search engine of software products (SPARS: Software Product Archive, analysis, and Retrieval System)
- Collaborative filtering
- Logical coupling
- GQM (Goal/Question/Metric) model

**Analysis Technology (1-1)
Code Clone Detection**





Analysis Technology (2) SPARS ~ Software Asset Search Engine

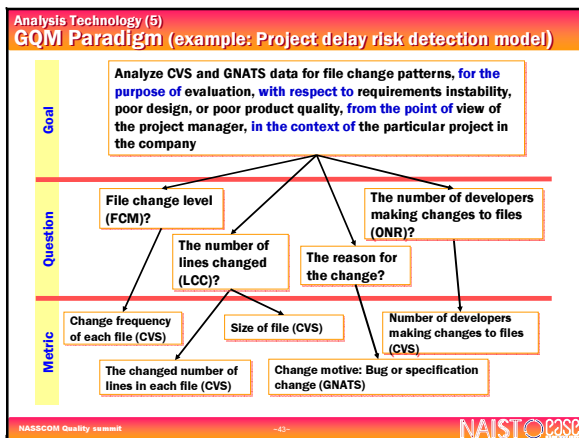
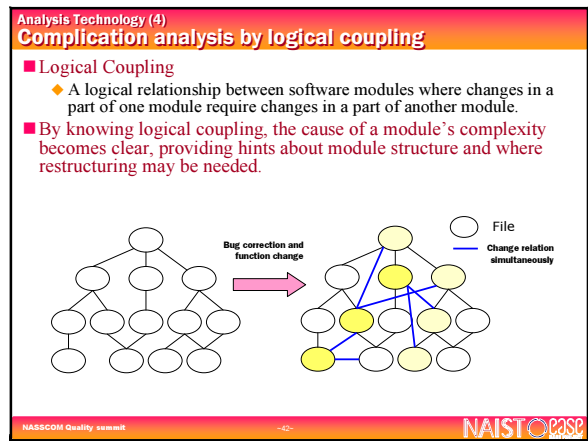
NASSCOM Quality Summit -39- NAIST CASE

Analysis Technology (3) Collaborative Filtering

- Robust estimation method with missing data
- Applicable to estimating various attributes of project/system from similar project/system profiles

	Focused	Representative	Q & M Resources	Collaborative	Outcome Adopted
App. A	9	9	9	7	7.5 (target)
App. B	8	7	8	?	8
App. C	?	8	8	8	7
App. D	7	6	?	9	6

NASSCOM Quality Summit -41- NAIST CASE

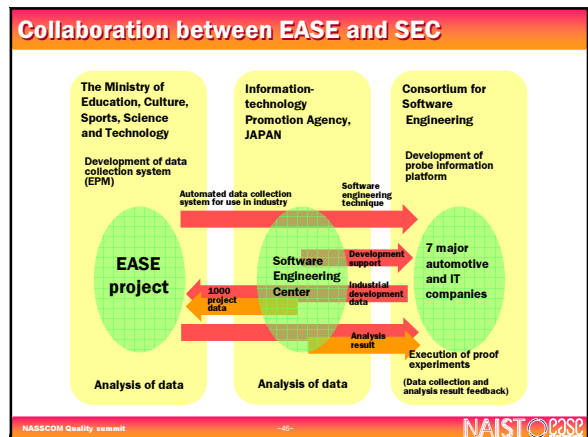
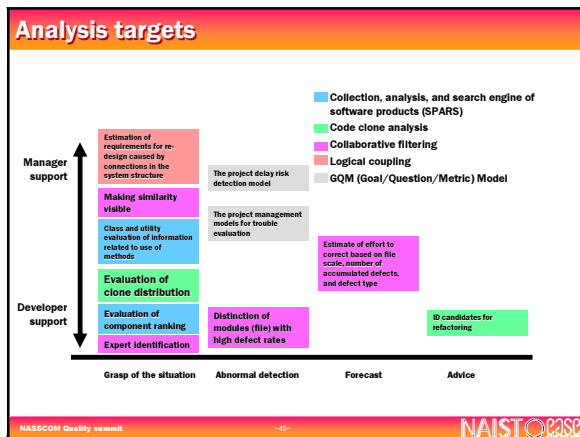


Analysis Technology (5) GQM Paradigm (example: Project delay risk detection model)

Goal

Analyze CVS and GNATS data for file change patterns, for the purpose of evaluation, with respect to requirements instability, poor design, or poor product quality, from the point of view of the project manager, in the context of the particular project in the company

NASSCOM Quality Summit -44- NAIST CASE



Applying EPM in Industries

- EPM is being applied to several real projects
 - ◆ Business systems (Hitachi GP, Ltd.)
 - ◆ Personal mobile applications (Mitsubishi Space Software Co., Ltd.)
 - ◆ Automobile information systems (SEC collaborative project)...
- Very low additional effort by developers for data collection
- Collected data is currently under analysis
 - ◆ Many findings only from collected data such as
 - Module refactoring candidates
 - Internal trouble detection

NASSCOM Quality summit -47- NAIST EASE

EPM application experience

Collaborating Enterprises	Hitachi GP, Ltd.	Mitsubishi Space Software Co., Ltd.
Software for application	Package software for business in municipality agency	Software purchased by a certain enterprise
Development language	Java, others	Java, others
Development period	6 months	10 months
Development scale	130,000 Loc	250,000 Loc
EPM introduction and operation cost	25 man-days	11 man-days
EPM subjective evaluation by developers	The automated data collection is useful. The presentation method of the analysis result is a useful means to see software and the development process objectively.	The data collection doesn't disturb the development. The project transitions can be objectively understood from the collected data.

NASSCOM Quality summit -48- NAIST EASE

Making of EPM open source

- Japanese version in June
 - ◆ Downloaded about 300 so far
- English version is available from TODAY
- License
 - ◆ Empirical Project Monitor License (EPML) that fixed the special contract articles to Common Public License (CPL) was made.
 - ◆ The user selects either CPL or EPML.

CPL
License regulations of U.S. IBM made based on IBM Public License. When software that combines the source code of CPL with the source code of original development is made, and the object code is distributed, it has to open only the part of the source code of CPL to the public.

EPML: Special contract of articles
The modification code need not be opened to the public, and be delivered to the project member the distribution ahead when the change to EPM and the source code (modification code) in an additional part are distributed or the individual who made the modification code distributes the modification code that hangs to the main employment corporation or individual moreover by the joint research project etc. to develop software by using EPM.
...

NASSCOM Quality summit -50- NAIST EASE

The CPL and EPML

CPL

Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, in any, and such derivative works, in source code and object code form.

Special contract of EPML articles

Notwithstanding the provisions of Section 3 (Requirements) of "Common Public License Version 1.0", in any of the following cases (hereinafter referred to as "use within the Project"), it shall not be required to publicly disclose or deliver to anyone to whom the source code of EPM has been distributed, any portion of such source code to which any change or addition has been made (hereinafter referred to as "changed or added code").
...

NASSCOM Quality summit -51- NAIST EASE

Lessons Learned in EASE

- The data collection cost is very low compared with the development cost.
 - ◆ It reached 10-20% of the development cost so far.
- Data analysis costs must be reduced.
 - ◆ As application experience increases, more systematic *reuse of the analysis work process* is expected.
- Clarification of concrete needs for analysis.
 - ◆ Example: "Analysis at the program module level and management of unexpected values are necessary."
- Grasp of the situation and detection of abnormalities can be done using only collected data and analysis results.
 - ◆ Even inexperienced students identified significant points and understood the software construction process more clearly using EPM data and analysis.

NASSCOM Quality Summit

--52--

NAIST EASE

Summary

Summary

- This talk was about
 - ◆ The current status of the software industry in Japan
 - ◆ The SEC (Japan Software Engineering Center)
 - ◆ Empirical software engineering
 - ◆ The EASE project (Empirical Approach to Software Engineering)
 - ◆ and EPM (Empirical Project Monitor)
- People don't like a document-based approach. (Phillip Johnson Dec. '99, Empirical software engineering.)
 - automated data collection system: EPM
- EPM: Simple is best.

NASSCOM Quality Summit

--53--

NAIST EASE

- Thank you for your attention.
- Please try the EPM open source software from
 - ◆ <http://www.empirical.jp/English/>

NASSCOM Quality Summit

--54--

NAIST EASE