Bridging the Gap between Business Strategy and Software Development

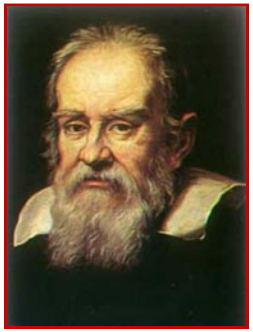
Victor R. Basili

University of Maryland
and
Fraunhofer Center - Maryland

Why Measurement?

"What is not measurable make measurable"."

Galileo Galilei



"A science is as mature as its measurement tools."

Louise Pasteur

"If you can´t measure it, you can´t manage it."

Peter Drucker

Why do Organizations Measure?

Understanding the Business

- Baseline models and relationships
- Critical factors affecting the business

Managing Software Projects Based on Quantitative Evidence

- Planning and estimating
- Tracking actual values versus estimates
- Decision-making

Guiding Improvement

- Baselining
- Prioritizing
- Assessing
- Packaging of Experience

Example Measurement Data

Resource Data:

- Effort by activity, phase, type of personnel
- Calendar time
- Total cost of ownership

Change/Defect Data:

- Changes and defects by various classification schemes

Process Data:

- Process definition and conformance
- Domain understanding

Product Data:

- Product characteristics
  - logical, e.g., application domain, function
  - physical, e.g., size, structure
- Usage and context information, e.g., design method used

Problems with Measurement

Problems

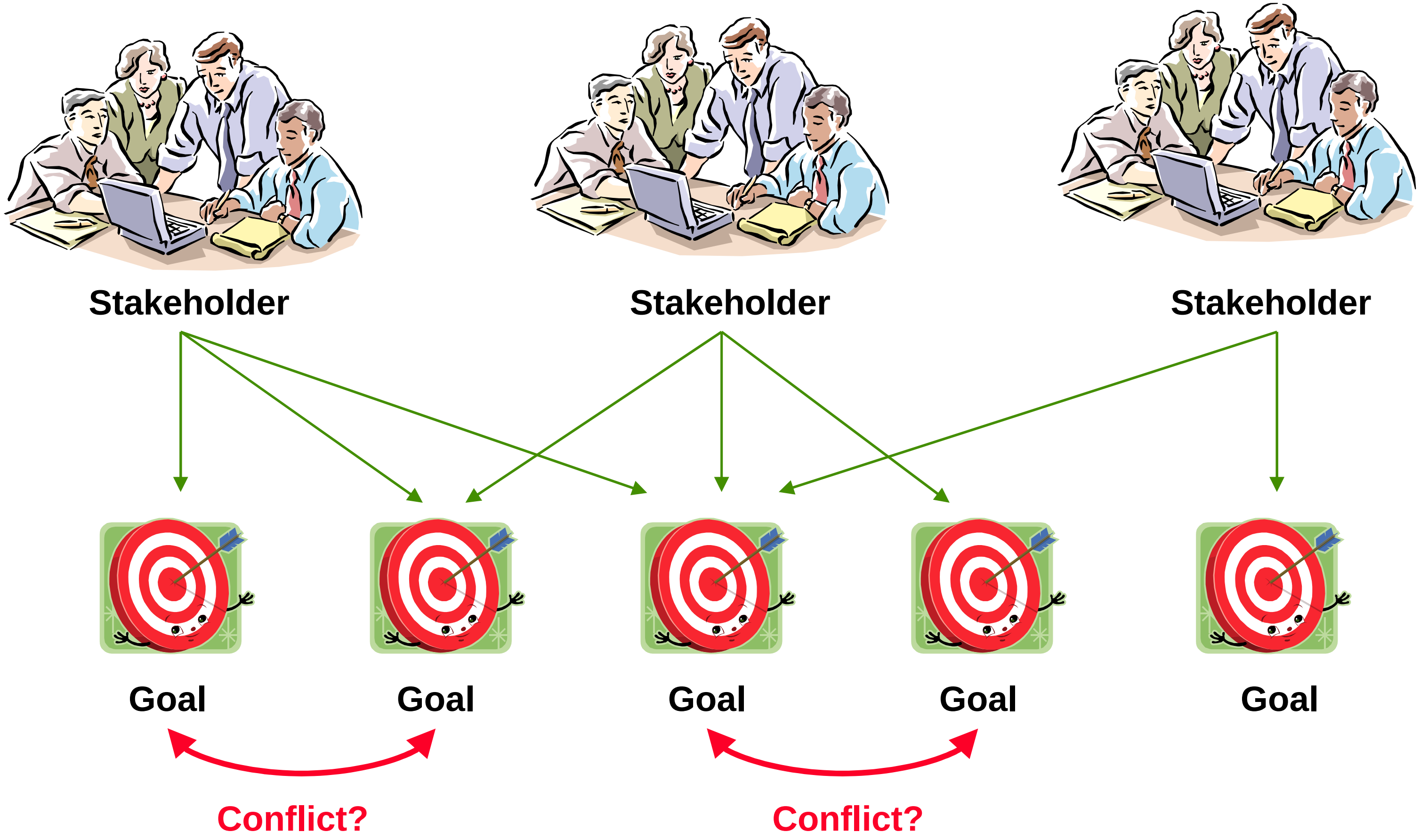
- Too much irrelevant data often collected
- Data incomplete, redundant, low quality or invalid
- More effort spent collecting data than analyzing data
- Data is not analyzed in the right environment
  - (no context or influencing factors considered)

Consequences

- Unnecessary effort
- Wrong conclusions can be drawn
- Discouraging for people collecting/analyzing data
- Not sufficient pay-off for the cost

⇨ **Goal-oriented Measurement**

Internal and External Stakeholders have Goals



Stakeholder

Stakeholder

Stakeholder

Goal

Goal

Goal

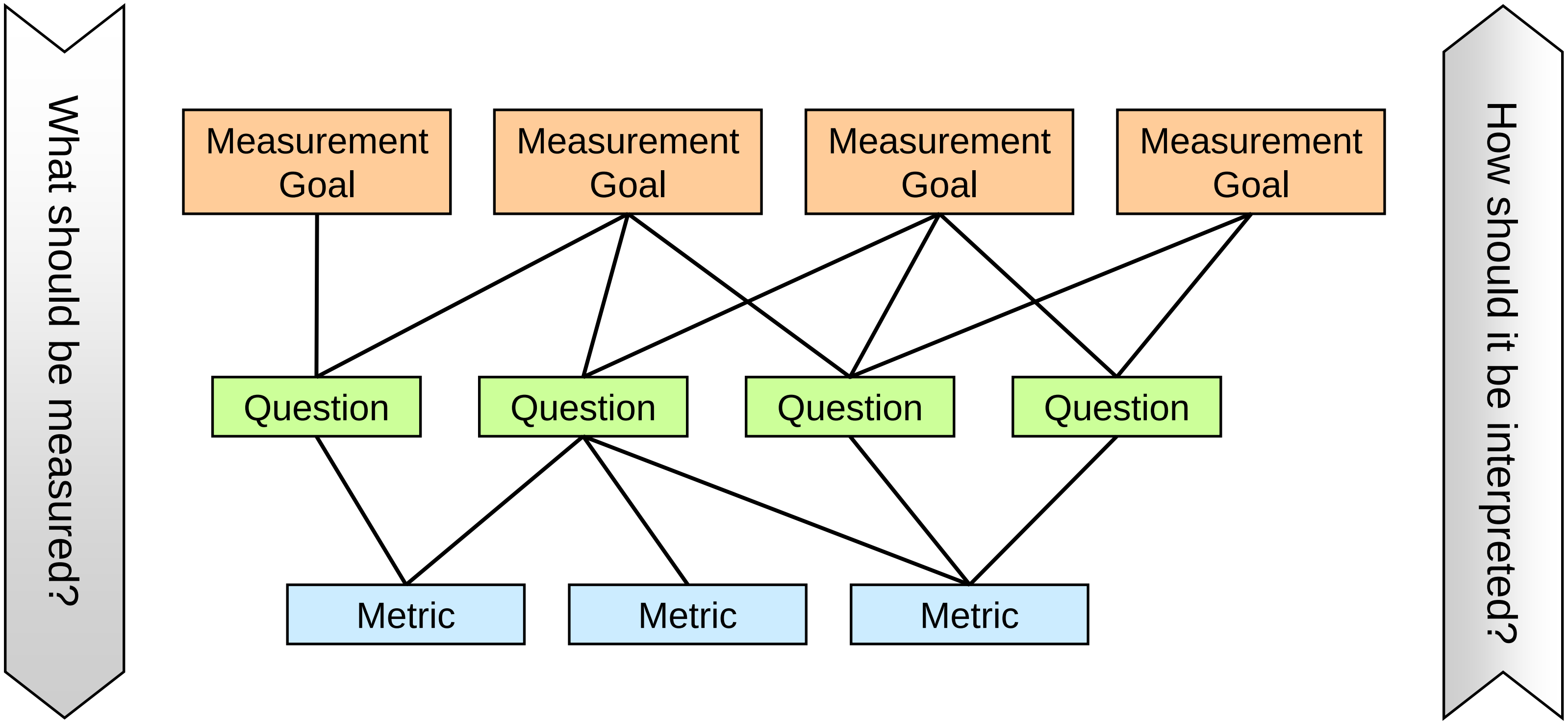
Goal

Goal

Conflict?

Conflict?

The GQM Structure



What should be measured?

Measurement Goal

Measurement Goal

Measurement Goal

Measurement Goal

Question

Question

Question

Question

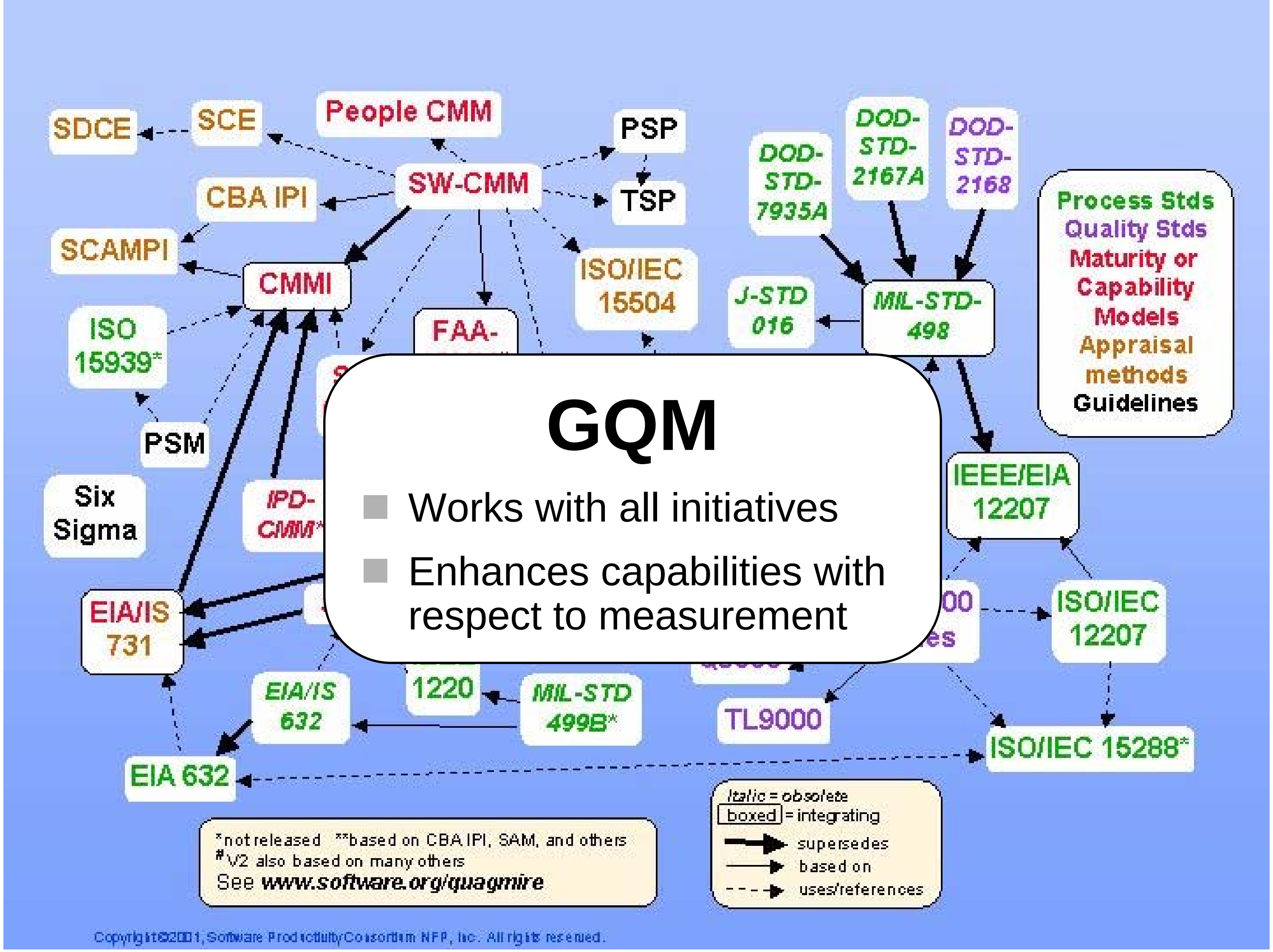
Metric

Metric

Metric

How should it be interpreted?

Measurement and other Quality Initiatives



*not released **based on CBA IPI, SAM, and others
#V2 also based on many others
See www.software.org/quagmire

Copyright©2001, Software Productivity Consortium NFP, Inc. All rights reserved.

GQM

- Works with all initiatives
- Enhances capabilities with respect to measurement

Generating a Measurement Goal

Consider the following situation:

Organization’s customers reporting too many failures and most of which should have been caught during system test (**Business Goal**)

It is considering adopting a new system test process (a risk and expense) and wants to try the new system test process on a pilot project to determine if it is doable and more effective than what it has been doing (**Software Goal**)

The organization has data on the number of faults identified by the system test process and the number released to the field for various products. It uses a waterfall type life cycle process, ... (**Context**)

To make an informed decision it must define the new test process, determine if it is being followed, characterize how well the process is identifying faults, and compare it to what they were doing before (**Measurement Goal**)

Template to Define GQM Goals and Example

	Description	Example
Object	Process, product, other experience model	Analyze **the system test process**
Purpose	Characterize, evaluate, predict, motivate, improve	for the purpose of **evaluation**
Focus	cost, correctness, defect removal, changes, reliability, user friendliness, ...	with respect to **defect slippage**
Viewpoint	user, customer, manager, developer, corporation, ...	from the point of view of **the corporation**
Context	Problem factors, people factors, resource factors, process factors, ...	in the context of the **specific organizational environment**

Measurement goals may be defined for any object, for a variety of reasons, with respect to various models of quality, from various points of view, relative to a particular environment

Goal/Question/Metric Approach

Defect Slippage Model

Goal:

Analyze the system test process for the purpose of evaluation with respect to defect slippage from the point of view of the organization …

Defect Slippage Model:

Let **Fc** = the ratio of faults found in system test to the faults found after system test on this project.

Let **Fs** = the ratio of faults found in system test to the faults found after system test in the set of projects used as a basis for comparison.

Let **QF = Fc/Fs =** the relationship of system test on this project to faults as compared to the average the appropriate basis set.

Goal/Question/Metric Approach

Interpretation of Defect Slippage Model

```
if QF > 1 then
    method better than history
    check process conformance
    if process conformance poor
        improve process or process conformance
    check domain understanding
    if domain understanding poor
        improve object or domain training
if QF = 1  then
    method equivalent to history
    if cost lower than normal then method cost effective
    check process conformance
if QF < 1  then
    check process conformance
    if process conformance good
    check domain understanding
    if domain understanding good
        method poor for this class of project
```

Goal Derivation Concepts we have Discussed so far

Business Goals: What the organization wishes to accomplish to maintain business success

Software Goals: What the software organization needs to accomplish to satisfy the organizational goals

Measurement Goals: Goals that can be measured and interpreted

Interpretation Model: A model that checks whether a measurement goal is achieved (and in consequence contributes to the achievement of related software and business goals

Context Factors: Environment variables that change the kind of models and data that can be used

Problems Establishing a Software Measurement Program

Building an Effective Software Measurement Program is difficult

It requires support for

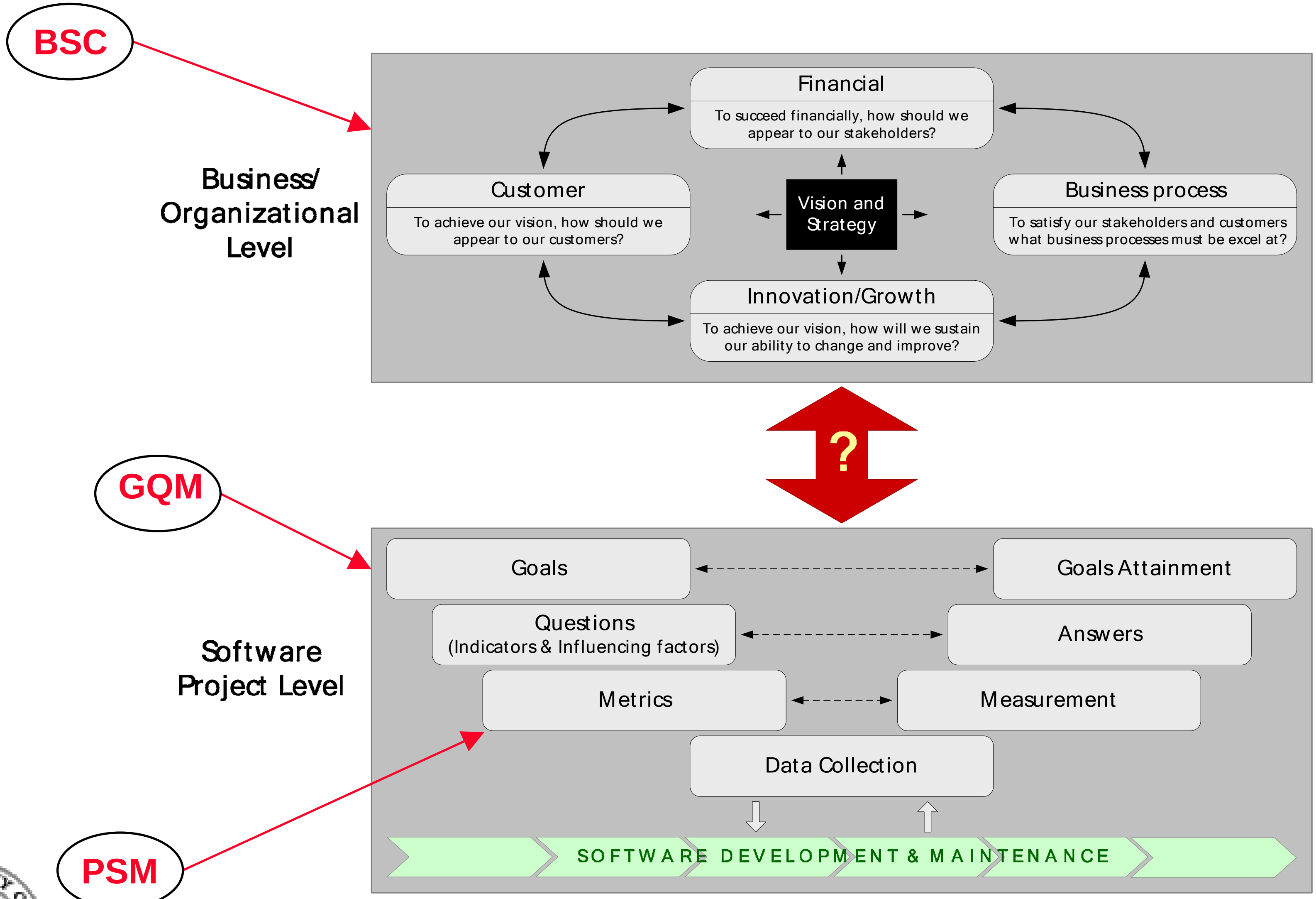
- **Defining and integrating** the organization’s top level corporate goals with its software goals
- **Mapping** the software goals to data, maximizing use of existing data
- **Evolving** the goals and data collected as the organization matures
- **Storing and retrieving** goals, data, and interpretations in context from an experience base

It involves **observation, experience facilitation, collaboration, decision making, analysis and synthesis** about goals, contexts, and assumptions

It assumes an **organizational structure** that sustains the process and learns

Most organizations fall short of putting together a successful program

Measurement Gap



BSC

Business/ Organizational Level

Financial

To succeed financially, how should we appear to our stakeholders?

Customer

To achieve our vision, how should we appear to our customers?

Vision and Strategy

Business process

To satisfy our stakeholders and customers what business processes must be excel at?

Innovation/Growth

To achieve our vision, how will we sustain our ability to change and improve?

?

GQM

Software Project Level

Goals

Goals Attainment

Questions (Indicators & Influencing factors)

Answers

Metrics

Measurement

Data Collection

SOFTWARE DEVELOPMENT & MAINTENANCE

PSM

Why do we want the connection?

Because:

Software **development and measurement activities should be justified** in terms of larger business objectives

Data from software **measurement must influence** higher-level decisions

An **analysis rationale** is needed that is meaningful at all levels

In order to:

clarify and harmonize goals and strategies

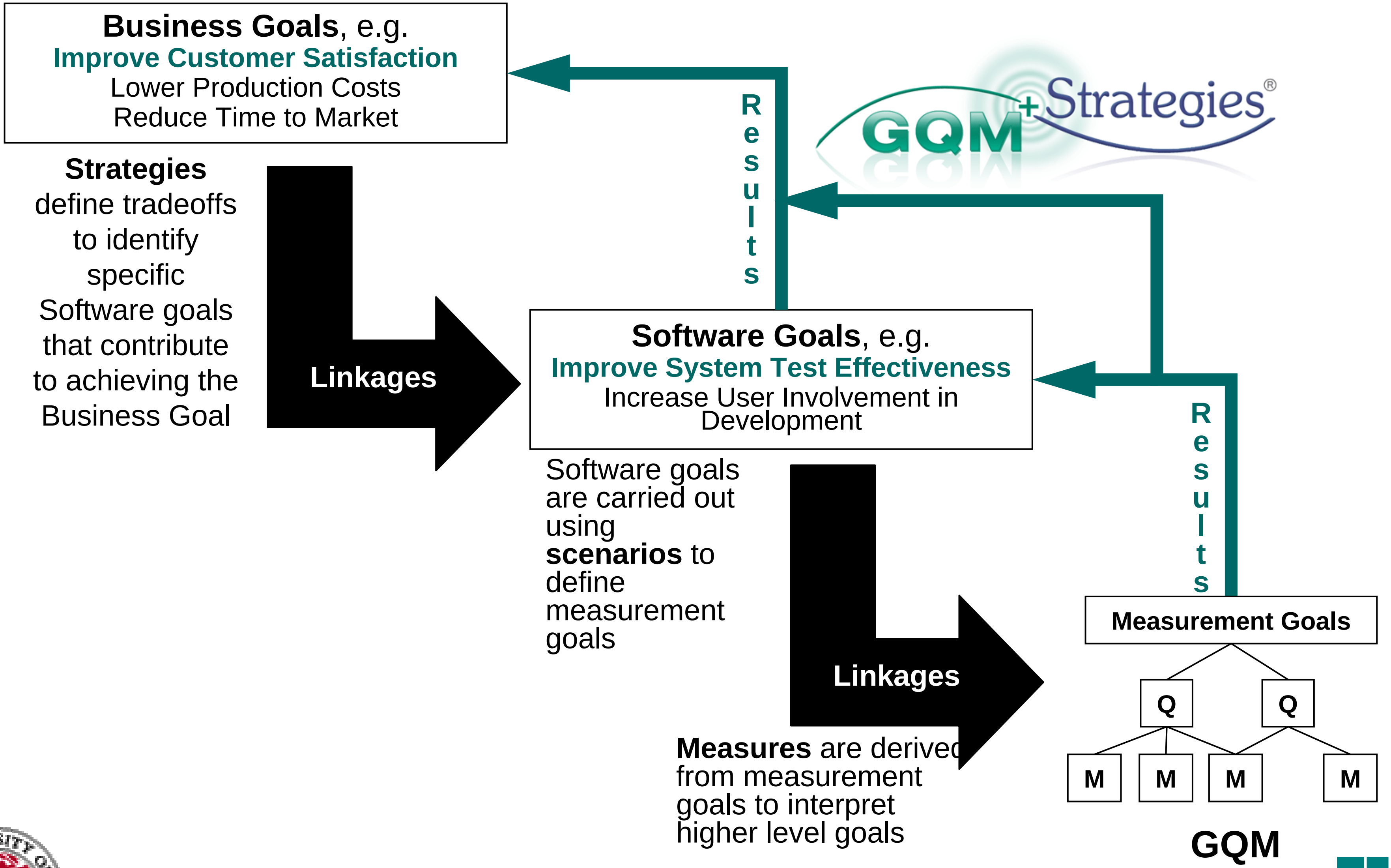
communicate business goals throughout an organization

align software goals to a strategy

monitor the deployment strategy

obtain feedback about a strategy and business goals

Closing the Measurement Gap



Business Goals, e.g.
Improve Customer Satisfaction
Lower Production Costs
Reduce Time to Market

Strategies define tradeoffs to identify specific Software goals that contribute to achieving the Business Goal

Software Goals, e.g.
Improve System Test Effectiveness
Increase User Involvement in Development

Software goals are carried out using **scenarios** to define measurement goals

Measures are derived from measurement goals to interpret higher level goals

Basic Business Goal Types

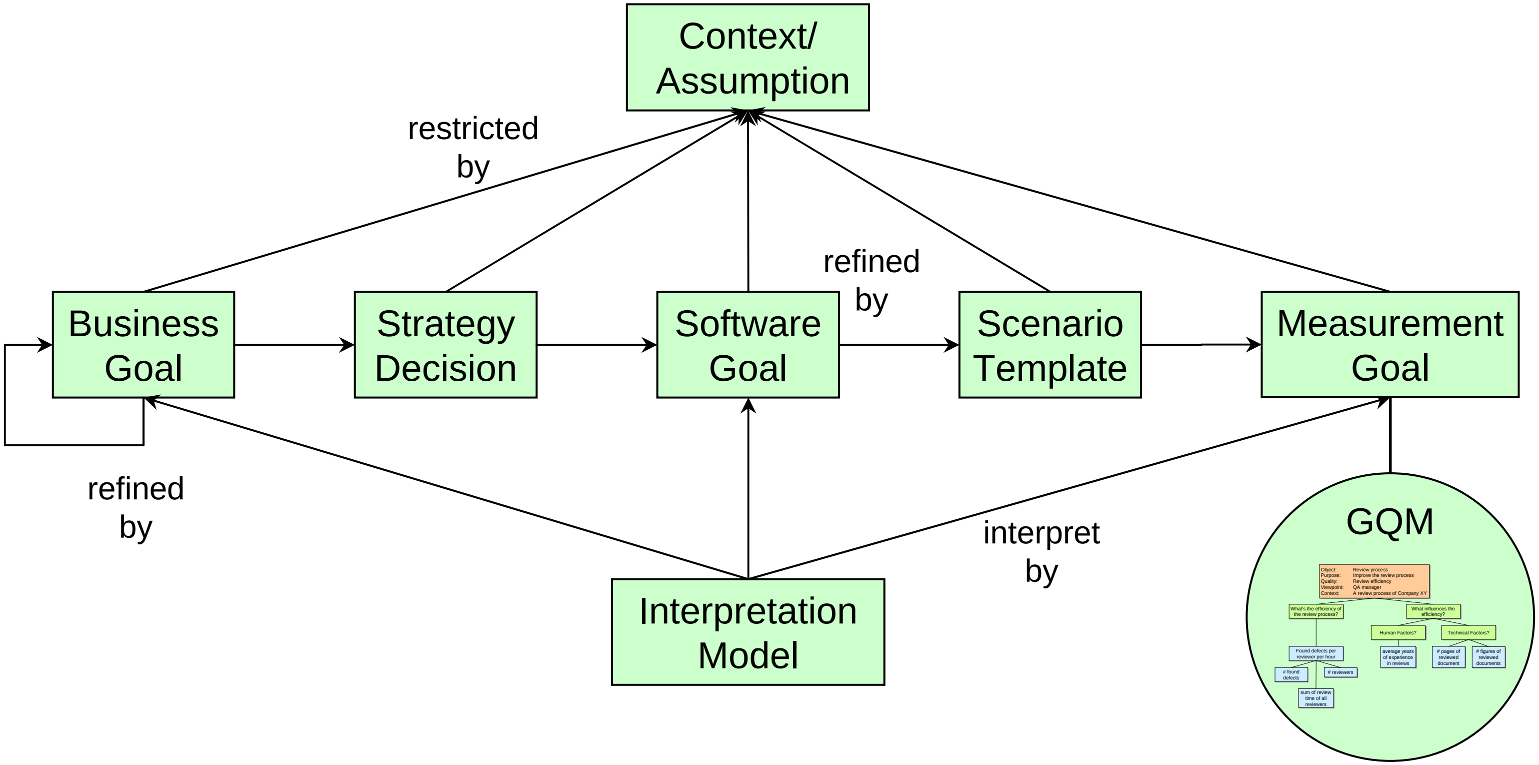
Growth goals: acquire new projects with current competencies areas; expand existing projects set; evolve existing competencies, build new competencies

Success goals: deliver good products to customers; control costs; shrink schedule; Increase profits; getting corporate visibility (awards etc.), building core competency

Maintain principle (internal) goals: transparency, employee satisfaction, controlled risk, learning environment
→ measure to assure no decrease

Specific focus goals: make helpdesk more efficient, predict if proposal effort has a good ROI

Defining The Right Goals
Goal Derivation Concepts



Context/ Assumption

restricted by

Business Goal

Strategy Decision

Software Goal

refined by

Scenario Template

Measurement Goal

refined by

Interpretation Model

interpret by

GQM

Object: Review process
Purpose: Improve the review process
Quality: Review efficiency
Viewpoint: QA manager
Context: A review process of Company XY

What's the efficiency of the review process?

What influences the efficiency?

Human Factors?

Technical Factors?

Found defects per reviewer per hour

# found defects

reviewers

sum of review time of all reviewers

average years of experience in reviews

# pages of reviewed document

# figures of reviewed documents

Goal Derivation Concepts

Step 1: Select the right business goals

Context

Highly competitive market for class of products

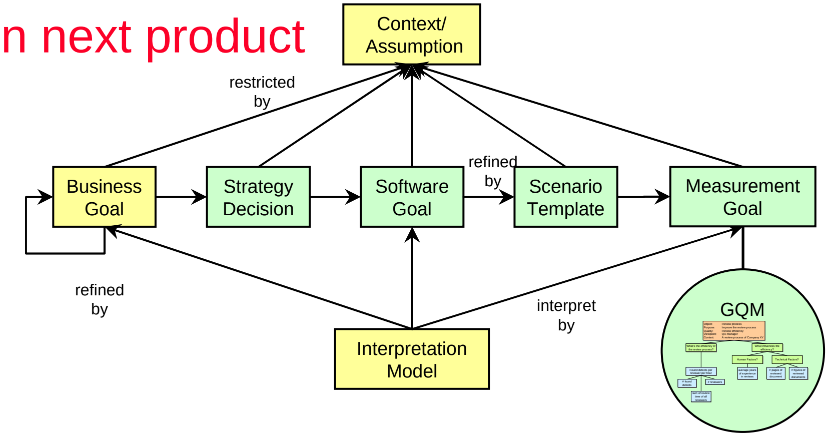
Assumption

Improving product will increase customer loyalty

Business Goals

Reduce next product time to market

Increase customer satisfaction on next product



Step 1: Formalize the Business Goals

Activity	Increase
Focus	Customer satisfaction
Object	Product “Splash”
Magnitude (degree)	10% reduction in number of customer complaints
Timeframe	12 weeks after release
Scope (context)	Web Products Division, Splash Project Manager
Constraints (limitations)	Splash price and functionality
Relations with other goals	Can conflict with development cost goals, schedule goals, …

Goal Derivation Concepts

Step 2: Select the right set of strategy decisions

Context

Little control over development process (too late)

There is limited budget for process improvement

Assumptions

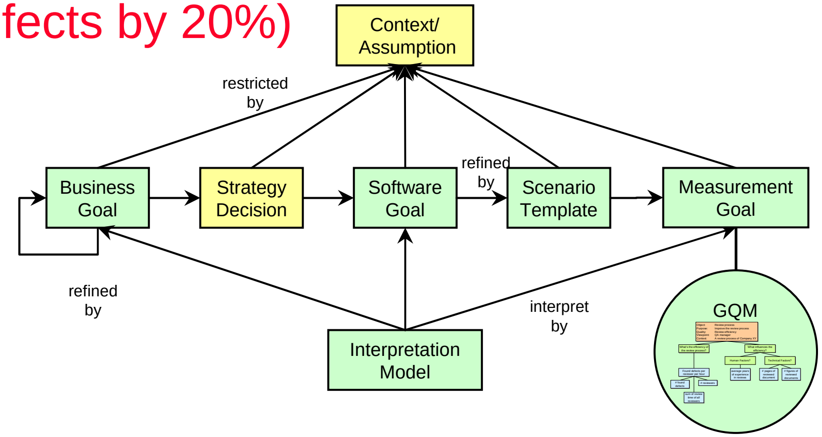
Customer satisfaction can be measured by # of customer complaints

Can’t make too many changes at once

Strategy Decisions

Build reliability in (e.g. implement fewer defects)

Test reliability in (e.g. remove defects by 20%)



Goal Derivation Concepts

Step 3: Select the right software goals

Context

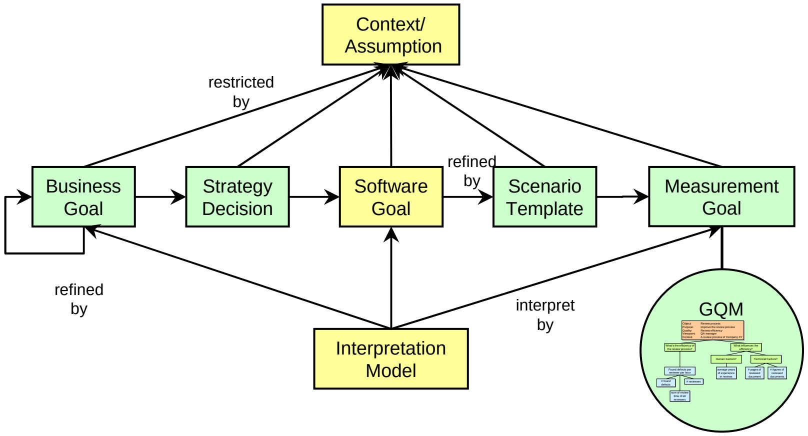
There is a system test process that seems appropriate for our context

Assumption

Can decrease # of customer complaints by 10% by reducing customer visible defects by 20%

Software Goals

Improve system test effectiveness by 20%



Step 3: Formalize the Software Goals

Activity	Decrease
Focus	Customer reported software defects
Object	System test process for Splash
Magnitude (degree)	Decrease customer reported defects by 20%
Timeframe	12 weeks after release (might check every week)
Scope (context)	Web Products Division, Splash Software Manager
Constraints (limitations)	Development cost and functionality
Relations with other goals	Can conflict with development cost goals, schedule goals, …

Goal Derivation Concepts

Step 4: Select the right scenario templates and steps

Context

Baseline data exists on defect slippage

Assumption

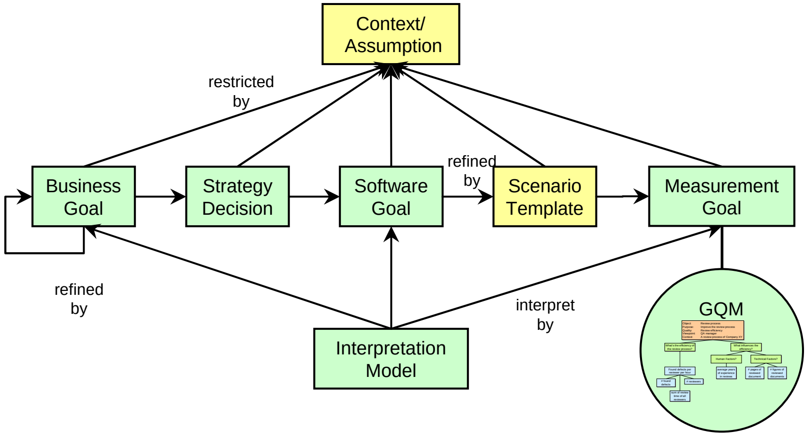
The projects that form the baseline are relevant to the current situation

Reducing defect slippage by during system test 20% will reduce customer reported defects by at least 20%.

Scenario Templates

Template A – based on historical data

Template B – based on hypotheses



Goal Derivation Concepts

Example Scenario Templates

Template A – based on historical data

Build a defect slippage baseline from historical data

Apply (Test) the new system test process and compare the defect slippage to past projects to evaluate its effect

Template B – based on hypotheses (no historical data)

Propose explicit hypotheses about defect slippage baselines based upon available expertise

Apply (Test) the new system test process and compare the defect slippage to past projects to evaluate its effect

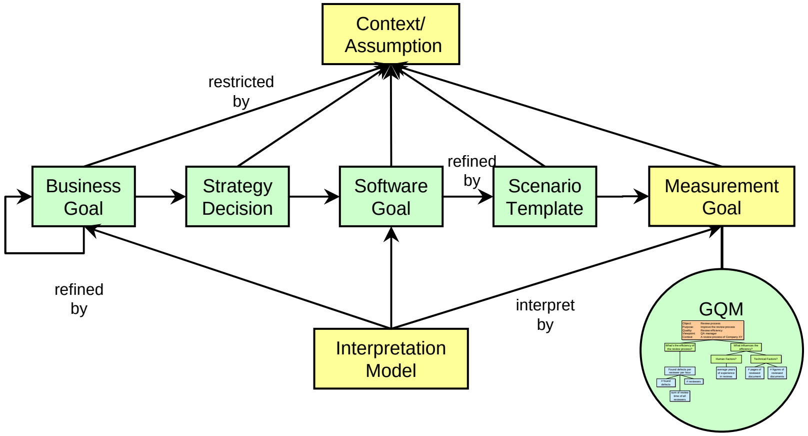
Goal Derivation Concepts

Step 5: Select the right measurement goals

A.1: Analyze **representative projects** in order to **characterize** them (build a baseline) with respect to **defect slippage** from the point of view of **the organization**

A.2: Analyze **pilot project using new system test process** in order to **characterize** it with respect to **defect slippage** from the point of view of **the organization**

A.3: Analyze **system test process** in order to **evaluate** it with respect to a **20% improvement in defect slippage compared to past projects** from the point of view of **the organization**



Step 5: Formalize Measurement Goals (A.3)

Object	System test process for Splash
Purpose	Evaluation
Quality Focus	20% defect slippage compared to prior projects
Viewpoint	Quality management
Context	Web Products Division

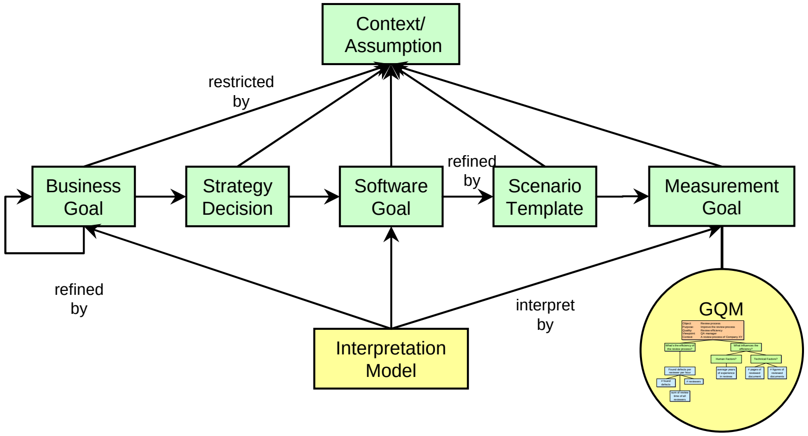
Goal Derivation Concepts

Step 6: Derive questions and metrics using GQM and interpret results

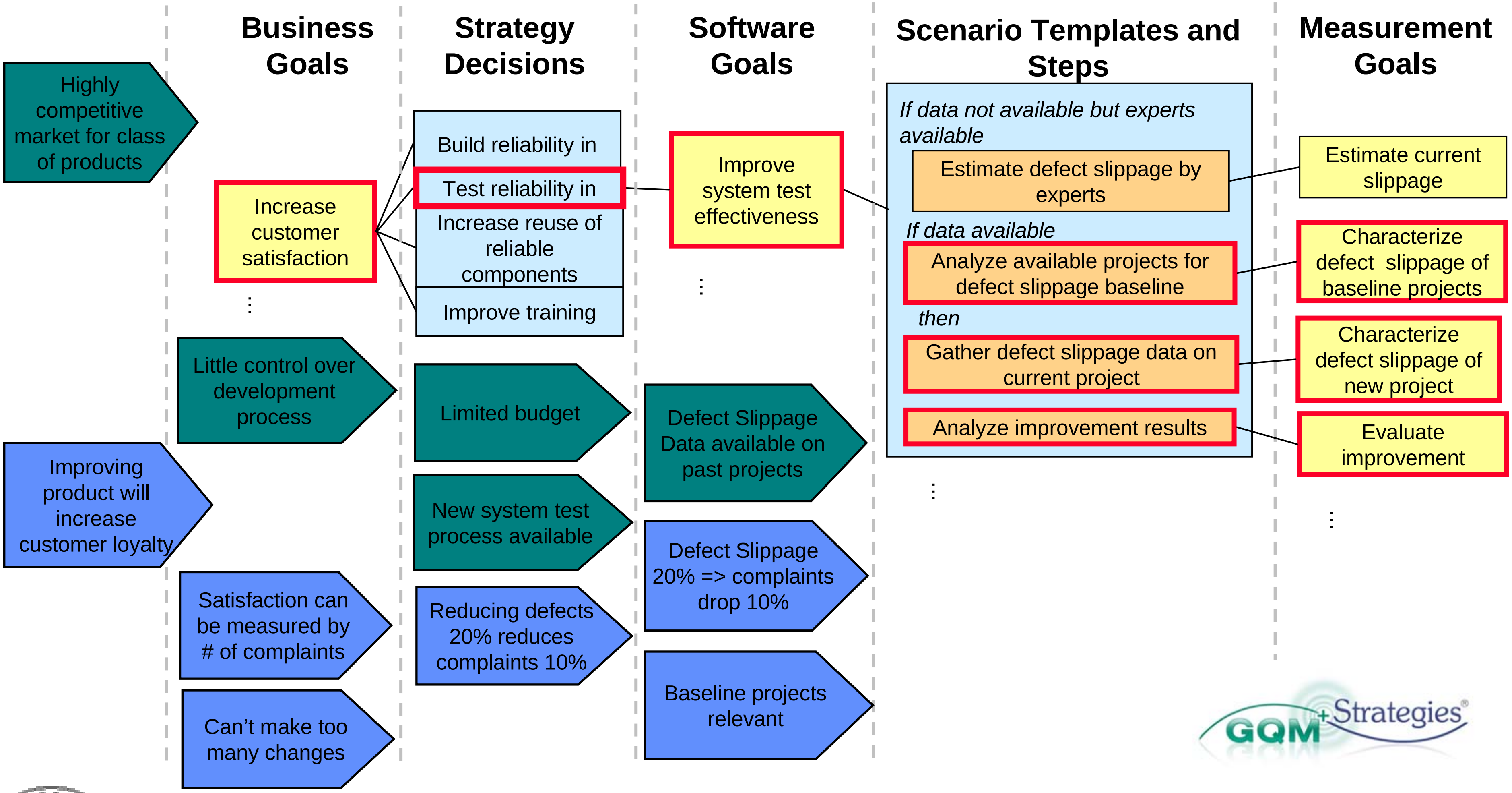
Apply the standard GQM approach (seen earlier)

Leads to the interpretation model (seen earlier)

Add interpretation for software goal and business goal



Business Goals to Measurement Goals



Business Goals | Strategy Decisions | Software Goals | Scenario Templates and Steps | Measurement Goals

Highly competitive market for class of products

Increase customer satisfaction

⋮

Build reliability in

Test reliability in

Increase reuse of reliable components

Improve training

Improve system test effectiveness

⋮

If data not available but experts available

Estimate defect slippage by experts

If data available

Analyze available projects for defect slippage baseline

then

Gather defect slippage data on current project

Analyze improvement results

⋮

Estimate current slippage

Characterize defect slippage of baseline projects

Characterize defect slippage of new project

Evaluate improvement

⋮

Little control over development process

Limited budget

New system test process available

Defect Slippage Data available on past projects

Improving product will increase customer loyalty

Satisfaction can be measured by # of complaints

Reducing defects 20% reduces complaints 10%

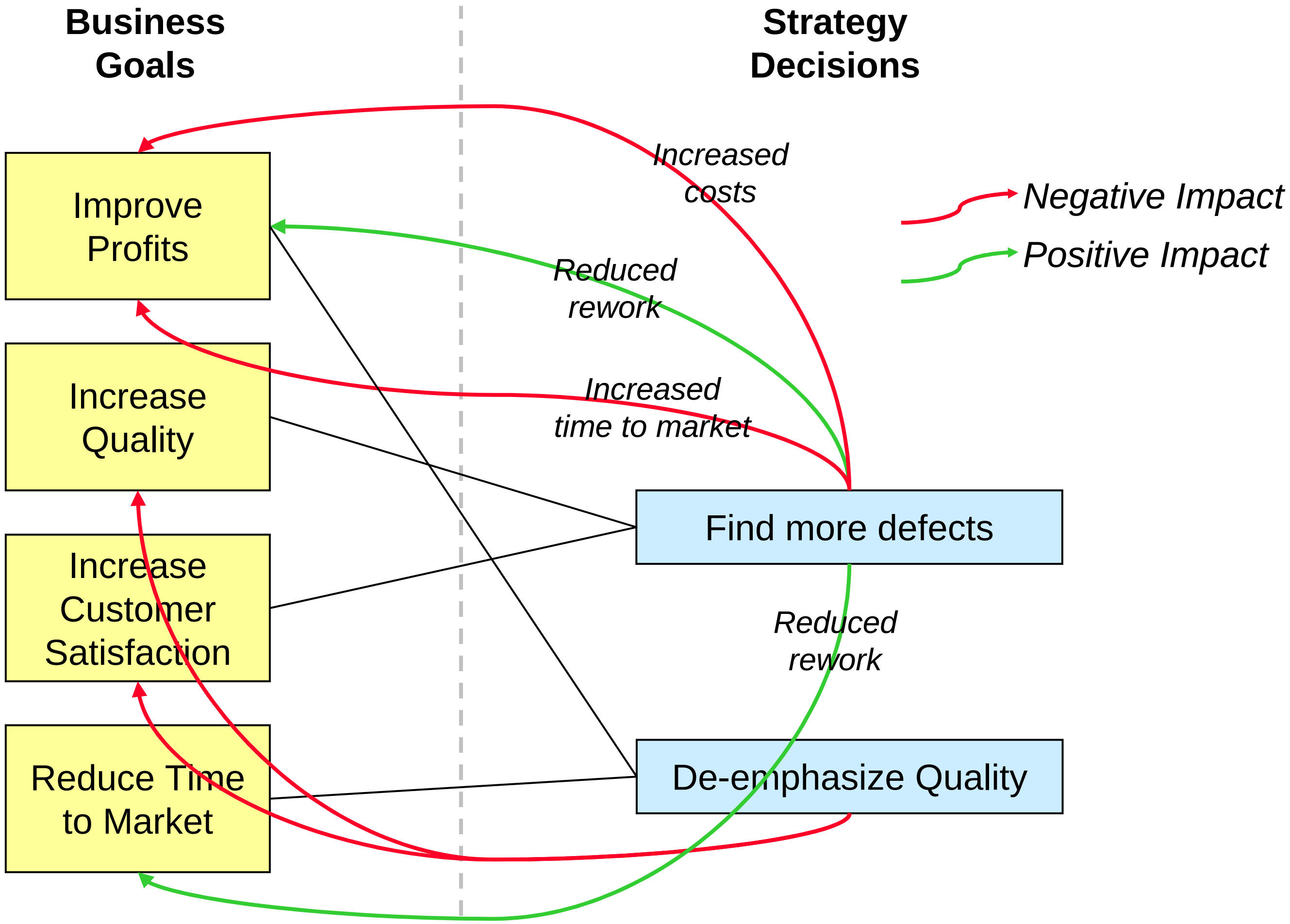
Defect Slippage 20% => complaints drop 10%

Can't make too many changes

Baseline projects relevant

Context / Assumptions

Goals may conflict with each other



Key Components to support the building of a software measurement program

An **experience base** of goals and scenarios that allow for the measurement program to be tailored to specific context variables and assumptions and is based upon experiences with various organizations

A **method** that takes into account the need for

- a goal hierarchy that allows goal choices for the needs of a particular organization and stakeholders
- dependency of goals on one another, e.g., temporal relationships
- strategies and scenarios for identifying sets of goals, recognizing which combinations of goals are needed depending upon environmental constraints
- mapping goals into existing data sets to maximize information while minimizing data collection
- the inheritance of data across multiple goals, i.e., mapping the data required from one set of goals onto others

An **expert** to help set up the measurement program in a the particular organization, including the generation of the goals, measures, data, and analysis

Summary

We've presented an approach to **linking** high-level business goals to operational level measurement goals that requires

- Defining **business goals**
- Choosing a **strategy** to achieve the business goal
- Translating the strategy, where appropriate, into **software goals**
- Choosing a **scenario** for achieving the software goal
- Deriving **measurement goals** from the scenario

The **payoff**

- Provides **justification** and **accountability** at all levels
- Provides a clear **plan of action**
- Provides guidance not just for planning, but also for analyzing and rolling up the resulting data **to the people who need to make decisions**
- Produces **reusable** artifacts

This is the work of

Dr. Victor Basili,
Dr. Jürgen Münch,
Jens Heidrich,
Dr. Mikael Lindvall,
Myrna Regardie,
H. Dieter Rombach
Dr. Carolyn Seaman,
Adam Trendowicz