2.4 Strategic Objectives

The repositioning of IT is framed and informed by strategic objectives. Guided by the strategic positioning of IT, these objectives are developed on the basis of the corporate strategy and business requirements.

The strategic objectives are binding – both on strategic IT management level and at operational level. They create the metrics by which you measure progress on implementation. The key elements of strategic objectives are:

- **IT goals**: The IT goals describe the status you are aiming for. They should be formulated such that you can measure the degree of goal attainment. Be sure to set goals with care: IT goals should be SMART, meaning Specific, Measurable, Achievable, Realistic and Time-bound. (There are also other terms for this common acronym, aimed more at the “people” aspect: Stretching, Motivating, Ambitious, Rewarding and Tangible, for instance.)
- **Principles**: Principles are specific, binding policies for action: make-or-buy preferences, for instance, or best-of-breed.
- **Strategies**: A strategy is the plan of action you decide on to achieve particular goals. As such, a strategy makes a statement on how goals are to be attained. Strategies in IT can include the application portfolio strategy, a strategy for sourcing or for innovation.

The following sections provide guidelines for deriving IT goals and best practices for principles and strategies.
2.4 Strategic Objectives

2.4.1 Deriving IT Goals

The process of deriving goals for IT from the overarching corporate goals is fundamental to strategy development. The connections between corporate and IT goals must be transparent, enabling you to verify that goals are consistent with corporate strategy. Figure 2.8 shows a best-practice derivation process; this is described in the following.

Fig. 2.8  Deriving IT goals

The IT goals take shape in an iterative process comprising two stages, Analysis & Identification, and Elaboration & Definition. What you do is this:

1. Look at the corporate goals and strategic positioning, identify and analyse what is driving the business. Put together a list of current business requirements.
2. Drill down the definitions of business drivers and requirements until you have a clear-cut statement that is specific enough to formulate “SMART” IT goals.

Helpful hint:

Break down business requirements along functional dimensions – business processes, products, business functions, business data and organisational structures. Analyse what interdependencies exist and what possibilities there are for IT support.
Example:

Corporate goal: to reduce the new-product development time by one third.
Step one in goal refinement: to reduce NPD time, the business process “research & development” and subprocess “series release” of the “production” business process must be investigated further.

3. Identify and document the requirements for IT: analyse all the IT assets in terms of dependencies and opportunities for IT support. These IT assets include the IT service and product portfolio, the application landscape, current technical standards, operating infrastructure, external and internal resources.

4. Refine and evaluate the requirements for IT, and use these requirements to formulate SMART IT goals. Costs and benefits should be your key evaluation criteria.

Important:

- In many cases the corporate strategy will not exist in written form. You will therefore have to make assumptions, document these assumptions and obtain feedback, for example, by staging workshops with representatives of senior management.
  It is crucial to have corporate goals documented explicitly. Without a clear statement on where the enterprise is headed, value-based management of IT will be impossible, because it will not be known what its benefit to the enterprise actually is!
- Maintain clarity on how corporate goals are cascaded into IT goals.
  An Excel spreadsheet is a good way to document the links and ensure connections can be traced back.

Example for Deriving IT Goals

ALPHA, a manufacturing company, has set itself the goals of increasing sales volume in product segment X in Europe by 20% and reducing service costs by 30%. These two corporate goals are described in Tables 2.3 and 2.4.

These goals have already been broken down along functional dimensions, i.e. steps one and two are not necessary here.
Table 2.3 Sales volume goal

<table>
<thead>
<tr>
<th>Corporate goal</th>
<th>To increase sales volume in product segment X</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context</td>
<td>An ease-of-use campaign to drive up European sales in product segment X</td>
</tr>
<tr>
<td>Content</td>
<td>To optimise product segment X to bring it more closely into line with European usability requirements and thus to raise sales volumes in Europe in product segment X</td>
</tr>
<tr>
<td>Impact</td>
<td>To become the biggest player in Europe in product segment X</td>
</tr>
<tr>
<td>Metrics</td>
<td>To raise sales volume in Europe by 20%</td>
</tr>
</tbody>
</table>

Table 2.4 Service cost goal

<table>
<thead>
<tr>
<th>Corporate goal</th>
<th>To reduce service costs in product segment X</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context</td>
<td>More economical delivery of services for products in product segment X</td>
</tr>
<tr>
<td>Content</td>
<td>To reduce on-site service calls by engineers in product segment X in Europe</td>
</tr>
<tr>
<td>Impact</td>
<td>To have service costs in Europe in product segment X comparable with competitors</td>
</tr>
<tr>
<td>Metrics</td>
<td>To reduce service costs by 30%</td>
</tr>
</tbody>
</table>

Step three is about seeking potential “quick-win” points for raising sales volume and reducing service costs in product segment X. The applications that support this segment are analysed, as are the current technical components. The investigation reveals a few opportunities: personalised self-service portals, for example, could increase customer loyalty, thus raising sales volume and reducing service costs. An analysis of the service and product portfolio shows up other options: it might be possible to introduce usage-based charging for software components and to expand products in segment X to include virtual reality software as fixed, configurable elements of products.

These opportunities are then evaluated by expected effort, benefit and various other criteria. The idea of a personalised self-service portal then takes shape as a specific proposal (see Table 2.5).

Important:

Refine your IT goals with enough detail to define metrics with specific values. For more information on metrics and performance measurement systems, please refer to Sect. 6.3.

By ensuring clear connections between IT goals and the metrics at operational level, it is easier to verify in the course of strategic controlling whether corporate goals are being achieved and business requirements fulfilled.
Table 2.5  IT goal: personalised self-service portal

<table>
<thead>
<tr>
<th>IT goal</th>
<th>Personalised self-service portal for product segment X in Europe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content</td>
<td>Provision of a portal infrastructure with easy user interface enabling end users to download new software statuses and information about products in product segment X</td>
</tr>
<tr>
<td>Impact</td>
<td>To be the first provider of self-service solutions in product segment X on European market</td>
</tr>
<tr>
<td></td>
<td>Less on-site service by engineers is required.</td>
</tr>
<tr>
<td>Metrics</td>
<td>I 30% reduction in on-site service in order volume</td>
</tr>
<tr>
<td></td>
<td>II At least 2,000 customers via self-service portal</td>
</tr>
<tr>
<td>Contribution to</td>
<td>Increase sales volumes in product segment X</td>
</tr>
<tr>
<td>corporate goal</td>
<td>Metric II has no direct correlation.</td>
</tr>
<tr>
<td>Contribution to</td>
<td>Reduce service costs in product segment X</td>
</tr>
<tr>
<td>corporate goal</td>
<td>Metric I has correlation with proportion of on-site service in service costs.</td>
</tr>
</tbody>
</table>

Fact file:

- There must be transparent connections between corporate goals and IT goals. Only then will you have evidence of IT’s contribution to business results.
- Break down IT goals to the extent needed to operationalise them.
- To monitor goal attainment, each IT goal must have metrics for which specific targets can be defined.

2.4.2 Principles for Strategic Guidance

Principles are a set of high-level statements which establish a general direction or a policy for IT-related decisions. An organisation might have principles to guide the selection of software solutions, for instance, project evaluation, roadmaps for IT landscape or software rollouts. Where principles exist, their application is essentially mandatory, and there must be good reasons for any non-compliance. Principles are in themselves enduring, they are not impacted by rapid pace of change in technology or products.

Note:

The term “guidelines” is often used instead of “principles”. However, guidelines do not have the same binding character as principles; as the name
suggests, they merely provide orientation. Whether you decide to apply principles or guidelines is very much contingent on the specific situation in your enterprise.

You document a principle by specifying its name, a brief description of what the principle is about, the requirements for applying it, the context in which it is used, and any notes on how to apply it (see Table 2.6).

<table>
<thead>
<tr>
<th>Principle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
</tr>
<tr>
<td>Reasoning</td>
</tr>
<tr>
<td>Prerequisites</td>
</tr>
<tr>
<td>Context of usage</td>
</tr>
<tr>
<td>Notes for usage</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2.6 Template for describing principles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Principle &lt;name of principle&gt;</td>
</tr>
<tr>
<td>Description &lt;Description of principle&gt;</td>
</tr>
<tr>
<td>Reasoning &lt;Explanation; i.e. why do we have this principle?&gt;</td>
</tr>
<tr>
<td>Prerequisites &lt;Prerequisites for application&gt;</td>
</tr>
<tr>
<td>Context of usage &lt;Which contexts are specifically included, which are excluded, e.g. restriction to organisational matters&gt;</td>
</tr>
<tr>
<td>Notes for usage &lt;Special notes on applying the principle, e.g. essential follow-up activities&gt;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2.7 Make-or-buy principle, example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Principle preference for standard software (make or buy)</td>
</tr>
<tr>
<td>Description</td>
</tr>
<tr>
<td>Reasoning</td>
</tr>
<tr>
<td>Prerequisites</td>
</tr>
<tr>
<td>Context of usage</td>
</tr>
<tr>
<td>Notes for usage</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2.7 Make-or-buy principle, example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Principle preference for standard software (make or buy)</td>
</tr>
<tr>
<td>Description</td>
</tr>
<tr>
<td>Reasoning</td>
</tr>
<tr>
<td>Prerequisites</td>
</tr>
<tr>
<td>Context of usage</td>
</tr>
<tr>
<td>Notes for usage</td>
</tr>
</tbody>
</table>

In deciding how to proceed on make-or-buy issues, you would normally consider questions such as competitive differentiation, whether the organisation has the resources and capabilities to develop software in-house, and what sourcing strategy it operates. A common statement is: standardisation where possible; variation where necessary.

Principles are essential for framing your decisions on IT assets and on strategic and operational IT management tasks. Here are a few examples of widespread and/or tried-and-tested principles (see also [Boa99], [Bur04], [Der06], [War02], [Wei04]) and others:
• Selecting software solutions:
  – **Best of breed (cherry-picking):** The Best-of-breed approach entails selecting the best software solution or product for each area of application according to a predefined raft of criteria. The “cherry-picked” solutions and products then have to be integrated into a unified entity.

  **Important:**
  The risk of the best-of-breed principle is that you can end up with an unmanageable patchwork of products, a situation exacerbated by frequent release changes and the need to juggle a variety of vendors. Therefore, introducing best-of-breed solutions can generate significant workloads for adjustment of interfaces.

  – **Make or buy:** The make-or-buy principle determines the organisation’s preference for bought-in software or own developments (see example in Table 2.7).

• Guidelines for selecting and appraising projects:
  – **Prioritise core business:** Projects focusing on securing core business are prioritised.
  – **Infrastructure projects first:** If a software project will entail major changes to IT infrastructure, the infrastructure adjustment must be managed in a separate project upstream of the software project.

• Design principles:
  – **Avoid heterogeneity:** Consolidate redundant technical components and applications through technical standardisation and homogenisation (see Sect. 5).
  – **Technical structure follows business structuring:** Applications should be organised by business criteria such as business components or functional domains. Related principles include “clarity of data ownership”, “clarity of process ownership” or “probability of change”.
  – **Avoid redundancy:** Effort should be taken to avoid redundancy in IT support and in technical standards. This means not having more than one application performing the same business function, and having a single lead system to manage master data.
  – **Only the lead system can modify master data:** Be sure to define a lead system for master data. The master data may only be modified by this system.
  – **Single point of distribution for master data:** Master data which is required by more than one system must be distributed by a single, central point of distribution, e.g. via a master data hub.
• **Procedures and principles in strategic IT planning:**
  
  – **Divide & conquer:** Break problems down into smaller units to resolve separately. Once you have your subsolutions in place, you can reassemble them into an integrated entity.
  
  – **Tuning:** Optimise and stabilise the present application landscape before investing in new solutions. For every new application you are considering, investigate whether it might be more opportune to expand or modify the resources you already have in place.
  
  – **Housekeeping** (also known as the survive strategy, opportunistic development or quarantine approach): The principle here is to survive with the IT landscape as it is. This means a halt to development, or reducing to a minimum all evolutionary development or expansion of the existing IT landscape. The focus is emphatically “must-have” development only.

  The IT landscape should be kept alive by quick-and-simple devices such as culling and consolidating master data, improving IT service organisation accessibility, centralising IT procurement, reviewing licences and maintenance contracts, bug patching in operational applications and making sure financial reporting can be completed as required.

  This principle is commonly applied to road-test new business models. For instance, if it is uncertain whether the business model will work, the organisation will decide to postpone fresh investment in a new target landscape until it can see its way ahead.

  – **One IT approach:** This approach entails consolidating the application landscape by introducing an all-in solution (bought-in or developed in-house) for a particular business segment. The “One IT” solution must adequately deliver the core functionality required for the business in question.

  This principle is commonly applied – albeit in modified form – in mergers and acquisitions. The IT solution of the stronger partner is wrapped around the acquisition target, and the data of the acquisition target is migrated to the selected IT solution. This generally means master data has to be consolidated. The One IT approach is also much in evidence driving decisions to buy standard software, the idea being to “stretch” the software to fit as many business segments as possible.

  – **Tried and tested over new:** Business systems with a proven track record should be favoured over new systems.

---

**Important:**

  • Rolling out a new application tends to take considerably longer than changing the existing landscape, and there is greater implementation risk (particularly in migration). Such risks can only be justified by the business perspectives afforded by the greater design freedoms.
You are unlikely to find a one-stop solution for all your needs. Even if you elect to use standard software, you will often need to make adjustments – in effect compromising the “One IT” principle.

In general, product rollout projects of extended duration will pursue a “survive” strategy.

- **Replacement strategy** (also renewal strategy): This entails a deliberate push to replace legacy systems.

- **Approaches for introducing new applications:**
  - **Big-bang**: New applications are introduced in one go, without intermediate stages. This is a way to make sweeping changes rapidly, but there is a greater risk of things going wrong.
  - **Evolutionary approach**: New applications are introduced incrementally. System renewal takes place step by step, with introducing intermediate solutions. This approach entails more effort overall than a big-bang rollout, but keeps the implementation risk down.

- **Increasing technical quality:**
  - **Flexibilizing**: Development of the business systems landscape over time focuses on keeping things flexible – by orienting development to components and services, ensuring integration ability and standardisation.
  - **Deconvolution**: Systems should be organised according to business criteria into loosely coupled components. The reasoning here is that convoluted connections between components and a lack of modularisation can make conducting projects in mature landscapes a highly fraught process.
  - **Decoupling**: Systems and loosely dependent components are decoupled via a broker or ESB.
  - **Encapsulation**: Specific application areas are segregated by introducing interfaces and standardisation.

- **Efficiency in operation:**
  - **Virtualisation**: Operating infrastructure units can be scaled using virtualisation technologies.

**Important:**

With enterprises being so different in terms of business models, corporate cultures or strategic direction, there is no single “right” way to do something. Select the principles which best fit your requirements. You should define principles for all IT assets and for all IT management tasks.
2.4.3 Strategies to Underpin Goal Achievement

A strategy is the plan of action you decide on to achieve particular goals. As such, it makes a statement on how goals are to be attained. Strategies can be aggressive, moderate or defensive.

Strategies serve to guide and limit IT decisions. As a rule, a strategy is implemented in the context of a project. You can define strategies to support the decision-making process pertaining to each of your IT assets. For more information, please refer to [War02].

Important:

Principles and strategies must all be clear and comprehensible. They must be articulated and codified. As a rule, it is better to apply just a few selected core principles and strategies. By keeping the number down, you will be better able to measure progress towards goals and steer a clear direction in implementation.

The following types of strategy are discussed in relevant literature (including [Ber03-2], [Boa99], [Nie05], [Rüt06], [War02] and [Wei04]):

- **Technical standardisation strategy**: A technical standardisation strategy is likely to frame definitions of standards for the technical implementation of software, hardware or network components. This strategy helps you reduce costs and avoid uncontrolled proliferation of components and systems in the IT landscape. See Sect. 5.

- **Application strategy**: An application strategy defines a framework and guidelines for deploying applications effectively and efficiently to support the business. The strategy guides and informs actions for adding value (e.g. software solutions to promote sales, or increase customer loyalty through CRM), for enhancing day-to-day operations and making the enterprise more agile (e.g. through introducing service-oriented architectures).

- **Innovation strategy**: An innovation strategy determines the enterprise’s policy toward new technologies. For example, the strategy might stipulate that new technologies are to be appraised proactively with a view to potentially introducing them into the enterprise. See Sect. 5.4.2.

- **Investment strategy**: Investment decisions must be consistent with the enterprise’s investment strategy. Decision criteria such as cost, benefit, strategy and value contributions are often visualised in a project portfolio.

- **Sourcing strategy**: The sourcing strategy answers the classic make-or-buy question: which IT services can or should an enterprise perform itself, and which should it source from other providers? A sourcing strategy appropriate to the needs of the business helps improve the cost structure, reduce operating risk, raise flexibility and give the enterprise more control over its IT.
• **Vendor strategy**: A vendor strategy helps the organisation select and manage its vendors. For instance, some organisations will operate a “preferred vendor” strategy, the idea being to avoid proliferation in the number of vendors, to reduce costs and secure vital expertise for the organisation.

• **Data strategy**: A data strategy essentially comprises guidelines and policies on data ownership, compliance, data protection and security, as well as statements on data quality requirements and how these are implemented.

• **Other examples**: Partner alliance strategy, integration strategy, open source strategy, network/hardware strategy and disaster prevention and recovery strategy.

There is considerable interdependence between these strategies – the technical standardisation strategy, for example, will determine how the application strategy is formulated.

**Important:**

Take care to select strategies consistent with your strategic positioning, IT goals, and the principles you are pursuing. Strategies must also be consistent with one another.

Given the importance of application strategies in strategic IT planning, we will now investigate various application strategies which have been tested and proven in enterprise use. You can choose the strategies which best fit your situation.

Application strategies can usefully be presented with portfolio diagrams. These give an at-a-glance overview of how applications are positioned relative to one another. Portfolio analysis is a method used in strategic management; portfolio management is a simple yet structured approach that helps the IT organisation roadmap its medium-term planning. It serves to categorise, evaluate, prioritise, cluster and manage control objects such as projects or applications, and create communicable units out of the resulting deliverables.

IT portfolio management is all about developing and strategically managing portfolios of items such as projects or applications. A portfolio creates a framework for appraising projects, applications or other control objects according to predefined criteria such as potential benefits, costs, strategy and value contribution, and risks, and for presenting them in diagram form.

Portfolios can be used to map the as-is status of the landscape, its to-be or planned status or both, according to various criteria. They help organisations develop planning scenarios to roadmap development of the IT landscape.

Portfolios improve the quality of communication between IT specialists and business management; as such, they are a vital tool for building acceptance of IT as a strategic partner. Portfolio management has gained widespread acceptance among
business managers, not least because it pulls together salient information in concise form – a welcome change for managers who are already inundated with data and information.

The following best-practice approaches for application strategies all have proven track records:

- Application strategy dependent on business process classification
- Application portfolio “strategy contribution and value contribution”, based on Ward/Peppard [War02]
- Application portfolio “technical quality and business value”, based on Maizilish/Handler [Mai05]
- Application portfolio “business value and technological appropriateness” (see [Buc05])
- Application portfolio “business value and risk”
  This portfolio has strong similarities with the portfolio “technical quality and business value” (see [Buc05])
- Application portfolio “strategic fit” and “landscape management fit” (see [Krc05] and [Krc90]). See Sect. 4. We are going to investigate the first four strategies in greater detail. For more information about the other best-practice approaches, please refer to the stated literature.

**Important:**

Portfolios are an excellent way to present strategies in visual form. The visual presentation is arguably one of the reasons why portfolio management has attained such overwhelming popularity in organisations. Portfolio diagrams do not inundate managers with information, and help present salient facts with at-a-glance visual clarity.

**Application Strategy Dependent on Business Process Classification**

The business processes are classified in a portfolio (see Fig. 2.9) according to their competitive differentiation and change rate. Business processes are considered to be high differentiators if they play a key role in sustaining or expanding the business. Business processes have a high change rate when they must constantly be adjusted to changes in the competitive environment or other external conditions.

Each quadrant denotes a typical strategy – in this example the four quadrants are titled “automated and integrated”, “agile”, “automated & standard” and “manual & standard”. Each application is placed in the quadrant containing the business processes which it supports.

Business processes that contribute to differentiating the organisation from its competition (these are placed in the “automated & integrated” and “agile” quadrants) help improve the position of the company on the market. Typically, these are
sales and marketing processes directed at acquiring new business, but can equally include processes to prevent customer churn and improve the enterprise’s image equity on the market. Call centres, for example, an Internet-based information system or a data warehouse can all contribute to raising the standing of the organisation in this way.

Differentiator-type business processes have to be as unique as possible – which typically entails using custom-developed software. Where such processes are subject to slower rates of change, they (and by association the interfaces between systems and partners) should be integrated and automated as tightly as possible. Sequences and procedures can be hardwired in this way, because the rate of change is relatively slow and the efficiency gains through automation will usually outweigh effort for making isolated changes. Such processes can build tightly-integrated connections with business partners, and the resulting boost to efficiency will reduce costs. This is the type of process in operation at organisations which integrate their suppliers into just-in-time manufacturing.

Differentiator-type processes with rapid change rates have to be designed so as to maintain agility of response to changes. A good example here is product management at telecom providers. These organisations set exacting benchmarks for their applications, needing systems which are flexible to configure and orchestrate. This requires component-oriented designs and flexible infrastructure such as SOA.
2.4 Strategic Objectives

Landscapes can consist of a blend of suitable bought-in and custom-engineered components.

Business processes such as payroll, accounting or HR management are less subject to rapid rates of change, and are less likely to be a factor differentiating the organisation from its competition. As such they belong in the “automated & standard” quadrant. These processes should be conducted as cost-efficiently and with as much standardisation as possible. Recurring tasks can be automated, as can business partner integration. This helps raise process efficiency.

When it comes to business processes which do not add competitive value, the enterprise should try to use standard software. Automation can be introduced to drive efficiency in processes which are not subject to frequent change. This can mean having hardwired processes. However, if more flexibility is required, overly complex IT systems are not the solution. What matters is to deliver basic operation for the main business transactions as cost-efficiently as possible. Other, nonstandard transactions have to be performed manually.

There are also alternative portfolio dimensions to classify processes (see [War02], [Wei04] and [Wei06]). Alongside differentiation and rate of change, other common classifications are strategy and value contribution, risk estimate and cost.

**Application Portfolio Matrix “Strategy Contribution/Value Contribution”, Based on Ward/Peppard [War02]**

This application portfolio is based on the concept proposed by McFarlan (see [War02]). It is in widespread use. Its concise presentation of how IT contributes to present and future business success makes it a good tool to aid communication between business managers and IT professionals on plotting the future direction of the application portfolio.

Like the model discussed above, this application strategy also maps the portfolio to a diagram. Applications are classified according to their value contribution and strategy contribution. The value contribution indicates the extent to which a system supports business processes which create competitive advantage for the organisation – e.g. in sales and manufacturing. The strategy contribution indicates how the system contributes to enacting corporate strategy, in other words, its part in the future business success of the enterprise. Section 6.3 provides further details on metrics.

The portfolio’s four quadrants (see Fig. 2.10) are described below:

- **High potential (“WILDCATS”):** Wildcats have the potential to play a pivotal role in the future business. Systems of this nature include tryout projects which are heavy on IT but short on actual business context (e.g. SOA infrastructure projects). With their benefit in terms of actual business contribution being so hazy, projects like this are as far as possible “quarantined” from the operational application landscape. It is also essential to keep close tabs on their costs. If costs run out of control or the project seems to be going nowhere, the organisation should have no compunctions about cutting the project short.
Strategic Planning of IT

Fig. 2.10  Application portfolio “strategy contribution and value contribution”

- **Strategic (“STARS”)**: Applications that contribute strongly both to value and strategy are the “stars”. They sustain both the present and the future business of the organisation. Through relentless business-oriented innovation, the enterprise can ensure these applications address strategic goals (see Sect. 5.4.2). Stars offer effective, efficient business support and have tight vertical integration (which means tying business partners into workflows).

- **Key operational (“CASH COWS”)**: Unlike stars, cash cows just keep the present business going. All business functions (billing, for instance) must work reliably, efficiently and securely. Risks should be avoided, which means no technical experiments.

  Cash cows are often the legacy software in the organisation. The question here is how much effort to invest in order to keep legacy software alive – do you want to invest defensively? You have to decide whether permanent revamping is worth the effort, or whether complete overhaul or new buy/build might be better.

- **Support (“DOGS”)**: Dogs are the applications that support non-differentiating business processes such as payroll or HR administration. These applications must be efficient and above all inexpensive in operation. Leverage whatever opportunities you can identify to streamline and automate these applications. New projects on these systems will usually only be undertaken when the organisation is forced to make changes on account of statutory frameworks or compliance requirements.

Table 2.8 lists questions to ask for classifying applications and allocating them to the most appropriate quadrant (see [War02]).

Figure 2.11 shows an example application portfolio. Alongside the classifying attributes “strategy” and “value contribution”, the chart also presents information about the cost and health of the applications.
### Table 2.8 Guidelines for classifying applications

<table>
<thead>
<tr>
<th>Application Feature</th>
<th>High potential</th>
<th>Strategic</th>
<th>Key operational</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Does clear competitive advantage accrue for the business through this application?</td>
<td>Yes¹</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Does the application support a specific business goal and/or a critical success factor?</td>
<td>Yes¹</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Does the application eradicate a business disadvantage of the organisation compared to competitors?</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Does the application prevent a foreseeable business risk from escalating into a more substantial problem in the near future?</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Does the application contribute to raising business productivity, or can it help achieve sustainable cost reduction?</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Does the application assist fulfilment of formal requirements?</td>
<td></td>
<td></td>
<td>Yes²</td>
<td></td>
</tr>
<tr>
<td>Can the application lead to future benefit or value for the company?</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

¹If the business benefit can be clearly articulated and it is understood how this can be attained; otherwise high potential.

²Where non-compliance could lead to significant business risk; otherwise support.

By adding planned applications, arrows, and an indication of which systems are to be replaced, IT managers can create a clear picture of development trajectories. Future applications have either “planned” or “to-be” status. Planned applications are those which are already framed by projects and have taken more specific shape. By way of contrast, a to-be application has not yet been planned so specifically — for instance, an organisation might have decided to introduce a new sales application, but has not yet determined whether to use standard software or develop its own.

**Useful hint:**

Assign a timestamp for each planning status of your application portfolio. This gives you a clear record (which you can trace back if necessary) of which as-is status (e.g. on January 1, 2009), which planned status (e.g. project portfolio on January 1, 2011) and which to-be status (e.g. roadmap 2015) were used.
Application Portfolio Mapping “Technical Quality” Against “Business Value”, Based on Maizilish/Handler [Mai05]

In this application strategy, the applications are classified into four quadrants according to technical quality and business value. The technical quality takes into account both the status of the application along its lifecycle and the extent to which performance, security or other quality requirements have been implemented. The technical quality largely determines what effort will be involved in evolutionary development of an application. The business value comprises the contribution which an application makes both to strategy and enterprise value (Fig. 2.12). For more information about these metrics, please refer to Sect. 6.3.

The integrated presentation of business and IT aspects makes apparent at a glance whether core business operability is in any way jeopardised. Which quadrant an application is placed in determines the strategy to be pursued in its future development:

- **Question marks**: Applications in this quadrant have strong technical quality but little business relevance.

  All applications in this quadrant are under observation, and the quadrant must be reviewed regularly to decide which applications should go, which should stay,
and which might appropriately be enriched with more business content, enabling them to be repositioned.

As a rule, the applications in this quadrant tend to be tryout-type applications which are heavy on IT, or sub-applications which have been rendered superfluous owing to changes to the business model. An example: reports generated for a purpose which is no longer current, and which have never been removed.

- **Invest and expand**: Applications in this quadrant are the archetypal “ideals”. Technical quality and business value are both high. Care should be lavished on these applications, with every effort taken to ensure they maintain their superior technical quality. Where possible and feasible, the applications should be enriched with more business functionality.

- **Renew**: These applications are of material importance to the enterprise’s core business, but their technical quality is low. Outage of these applications will jeopardise the enterprise’s business continuity. Accordingly, they urgently need modernisation or, if this is not possible, they must be replaced by other systems with high technical quality.

- **Divest**: Applications which are neither important for the current or future business nor technically viable should be divested, and their functionality shifted to other applications with better technical quality.
Beginning with the current application portfolio, the organisation can draft out its roadmap for developing the application landscape over the coming years (see Fig. 2.13). In accordance with the strategies the enterprise is pursuing, applications are then “divested”, “expanded”, “repositioned & expanded” or “renewed”.

Important:

An application portfolio can present just the applications which are already in productive operation; it can also include planned or to-be applications. Arrows, or a sequence of version portfolios (roadmap), can be used to map changes over time.

Application Portfolio “Business Value/Technological Appropriateness”, Based on [Buc05]

In this application strategy, based on [Buc05], applications are classified according to their business value and technological suitability (or appropriateness). Instead of four quadrants, this portfolio works with five areas.

The technological appropriateness dimension encompasses the technical quality, modernity (how “state-of-the-art” an application is), complexity and age of both the application itself and the operating infrastructure on which it runs (Fig. 2.14). It
also includes an appraisal of the application lifecycle and available IT resources.\footnote{Including specialist expertise such as Assembler programming.} For details on metrics, please refer to Sect. 6.3.

The relative positioning of applications within this matrix shows up the areas that cost reduction measures could feasibly target without jeopardising the overall sustainability of the applications.

Which quadrant an application is placed in determines the strategy to be pursued in its future development:

- **Catch up on investment**: The applications grouped in this area have high business value but inadequate technological appropriateness. Since these applications are critical for the enterprise, they must be monitored carefully. IT deficits must be mitigated or eradicated altogether; an alternative is to replace these applications by others with higher technological quality and equivalent ability to perform the business functions in question.
• **Skim benefits**: The applications in this area are appropriate in terms of both technology and the value they deliver to the business. Enterprises should fully skim business benefits of these applications, where feasible also rolling them out at other locations.

• **Balanced at present, continue**: The applications in the centre of the portfolio strike a good balance between business value and technological appropriateness. These applications can include IT systems for support processes such as accounts management or HR administration. Accordingly, no action need be taken pertaining to these applications.

• **Limit investment**: The applications in this area are technologically appropriate but have only low business value. Accordingly, investment should focus on functional extensions that raise business value.

• **Monitor, or invest and cull**: The applications in this area deliver only little support to the business and moreover are technologically inappropriate. Keep a close eye on any applications here – it may be appropriate to retire certain of them and introduce replacements.

Each of the application portfolios and associated information strategies discussed above has a proven track record in real business operation, having been applied successfully for strategic planning and management of the application landscape in multiple organisations.

**Helpful hint:**

At the initial, early stage in landscape appraisal, it is fine to use an essentially pragmatic approach to classifying applications. Organise a workshop with participants from business departments and IT, and have the participants classify the applications. The discussions this engenders, and the process through which stakeholders move toward consensus, are usually an efficient way to reach an agreement on classification.

In practice, this approach will seldom produce divided opinion!

**Fact file:**

- There are different types of strategy, e.g. innovation strategy or application strategy.
- There are numerous best-practice approaches for application strategies. You can use these to develop your own strategy customised to the concerns you wish to address.
- Portfolios are a useful tool to visualise strategies.
Strategic IT Management
A Toolkit for Enterprise Architecture Management
Hanschke, I.
2010, XI, 342 p., Hardcover
ISBN: 978-3-642-05033-6