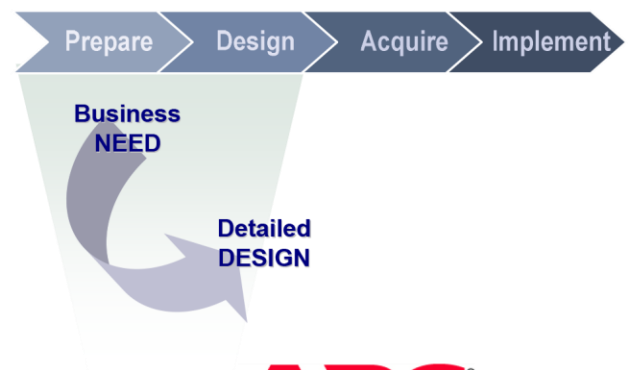


Data Center Projects: System Planning

By Suzanne Niles

White Paper #142



APC[®]
Legendary Reliability[®]

Executive Summary

System planning is the Achilles' heel of a data center physical infrastructure project. Planning mistakes can magnify and propagate through later deployment phases, resulting in delays, cost overruns, wasted time, and ultimately a compromised system. Much of the trouble can be eliminated by viewing system planning as a data flow model, with an orderly sequence of tasks that progressively transform and refine information from initial concept to final design.

Introduction

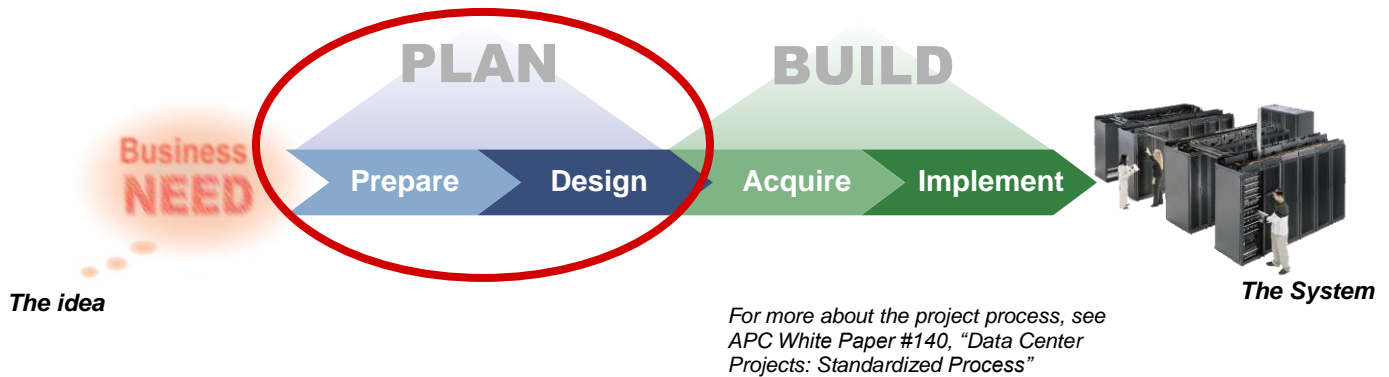
Planning remains a major challenge for small and large IT facilities. Data center build and upgrade projects are typically planned using methods resembling art more than science, in a process often perceived as intimidating, unstructured, and difficult. Plans are poorly communicated among the various business stakeholders within the organization. Planners may be presented with proposals that are presented in excruciating technical detail, yet still lack the information they need to make decisions. Small changes in plans can have major cost consequences or create downstream surprises and disasters.

Consulting engineers have traditionally carried the burden of system planning for data centers, with great care focused on the unique requirements and design of each project. This accumulated experience of decades of data center planning now offers an opportunity to consolidate that knowledge and evolve toward a more standardized approach that can provide increased quality and economies of scale for both provider and consumer.

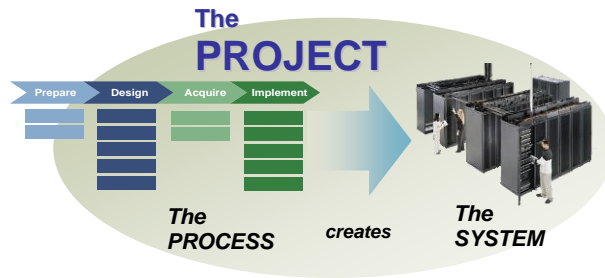
In a data center construction or upgrade project, it is the early planning activity that offers the greatest potential for mistakes. Most defects that turn up in the later stages of a project are caused not by problems in the physical components of the system, but rather by oversights or misinformed decisions during planning. Fortunately, it is possible to avoid many of these problems by having the right people making the right decisions in the right sequence.

Figure 1 shows where planning occurs in the context of a data center project. The **PLAN** portion of the project, consisting of the **Prepare** and **Design** phases, lays the foundation for everything that follows. Careful navigation through this part of the process is critical to project success. These two phases set up the details of both the physical **system** to be created and the **process** that will create it.

Figure 1 – The PLAN portion of the process lays the foundation for the project



Process planning includes such things as assigning responsibilities, configuring project management, making outsourcing decisions, budgeting, and scheduling. **System** planning – the subject of this paper – addresses the hardware configuration that is the tangible outcome of the project. System planning transforms the original project concept into a detailed construction design. To support the system planning effort, a clearly structured model of data flow, dependencies, and design strategy provides essential guidance. This paper does not offer detailed planning advice, but rather illustrates a framework for organizing, understanding, and tracking the logical sequence that carries the system design from concept to blueprint.



The planning sequence described in this paper covers the design of the data center’s **physical infrastructure** layer, whose job it is to house, power, cool, and protect the data center’s computing and network functions, which are the “IT layers” of the data center. This planning sequence assumes that the determination of computing requirements and the general design of the IT layers has already taken place. Certain characteristics of the IT system concept then provide essential input to the planning of the physical infrastructure, as will be seen in the first task of the planning sequence.

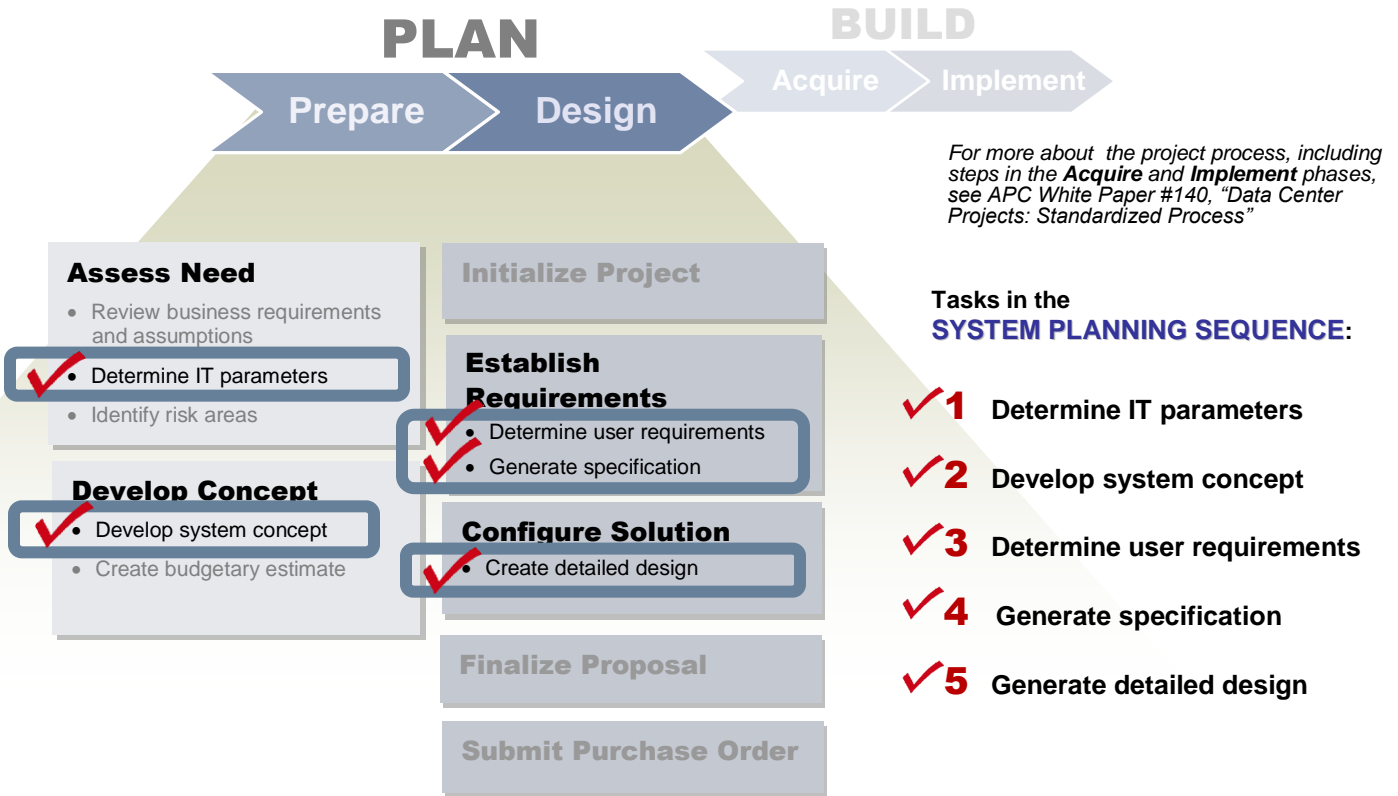
What is the “System Planning Sequence”?

The “system planning sequence” is the logical flow of thought, activity, and data that transforms the initial project idea into a detailed installation plan. In APC’s implementation of the standardized project process, system planning is sequenced into five tasks that take place during the **Prepare** and **Design** phases of the project.

These five system planning tasks occur within the context of other tasks necessary for the overall conduct of the project such as budget analysis, hiring of service providers, and proposal generation. However, the system planning tasks form their own logical sequence that can be extracted and considered by itself. When isolated like this, it becomes much easier to visualize and comprehend the flow of activity that transforms and refines a general idea into a detailed system design.

Figure 2 illustrates these five tasks that comprise system planning, both as they occur in the context of the **Prepare** and **Design** phases (circled and checked) and pulled out as a separate list of five items.

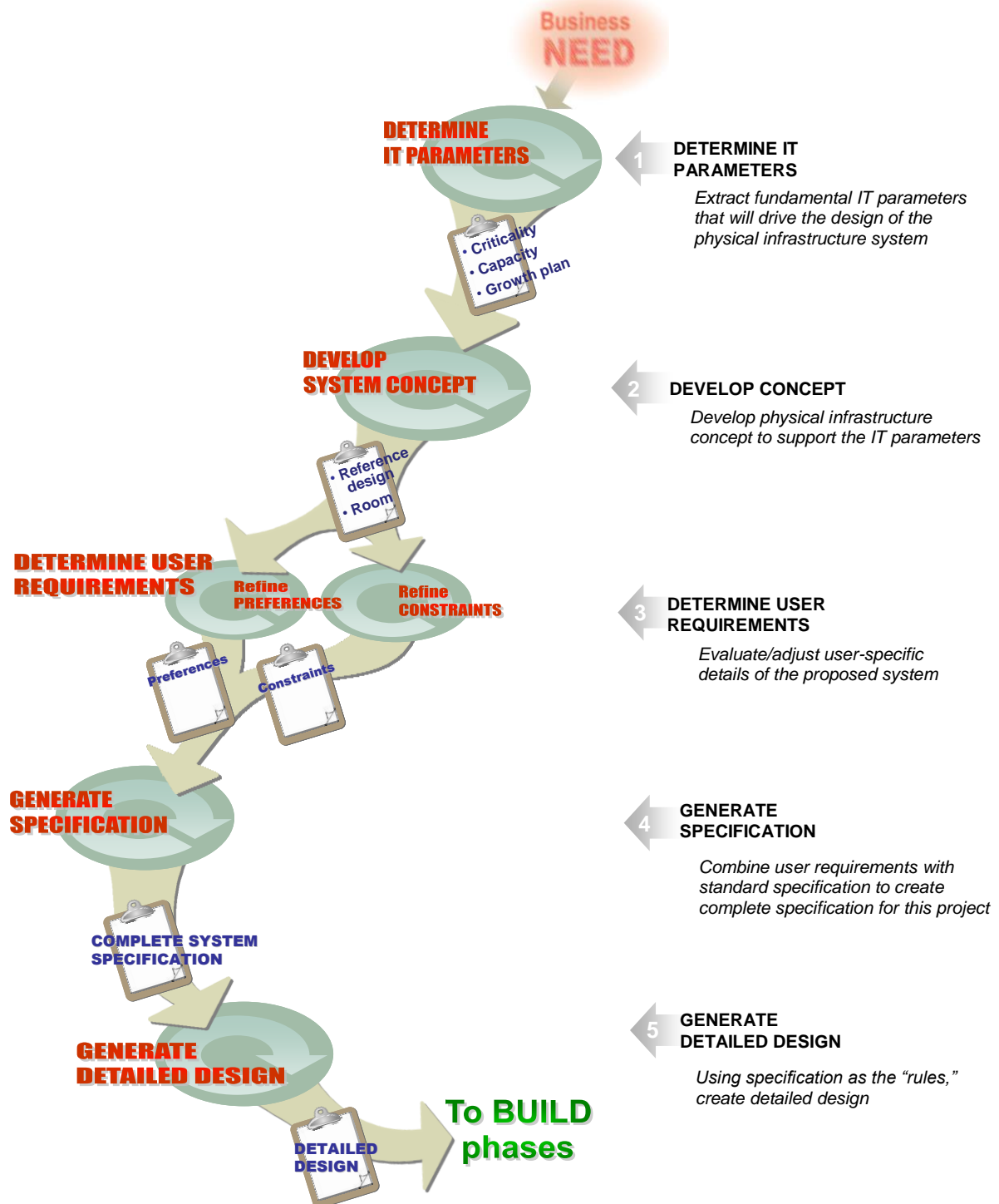
Figure 2 – The five tasks of the “system planning sequence”



The planning sequence as a data-flow model

Each of these five tasks takes information as input, transforms it or adds to it, and sends it along to the next task. This progression can be modeled using the data-flow diagram in **Figure 3**. The data that is flowing and being transformed is the developing description of the system. In this diagram, data is shown as pages on a clipboard traveling from task to task (the blue ovals), with a new page of data added on top at each stop along the way. The pages of data from previous stops travel along with the new data, so that all previous data is available for reference or review, if necessary, at any point in the sequence. In this way, the intentions of earlier planners are preserved, and downstream problems can be traced back to previous data and decisions for re-evaluation, if necessary.

Figure 3 – The five tasks of the “system planning sequence”

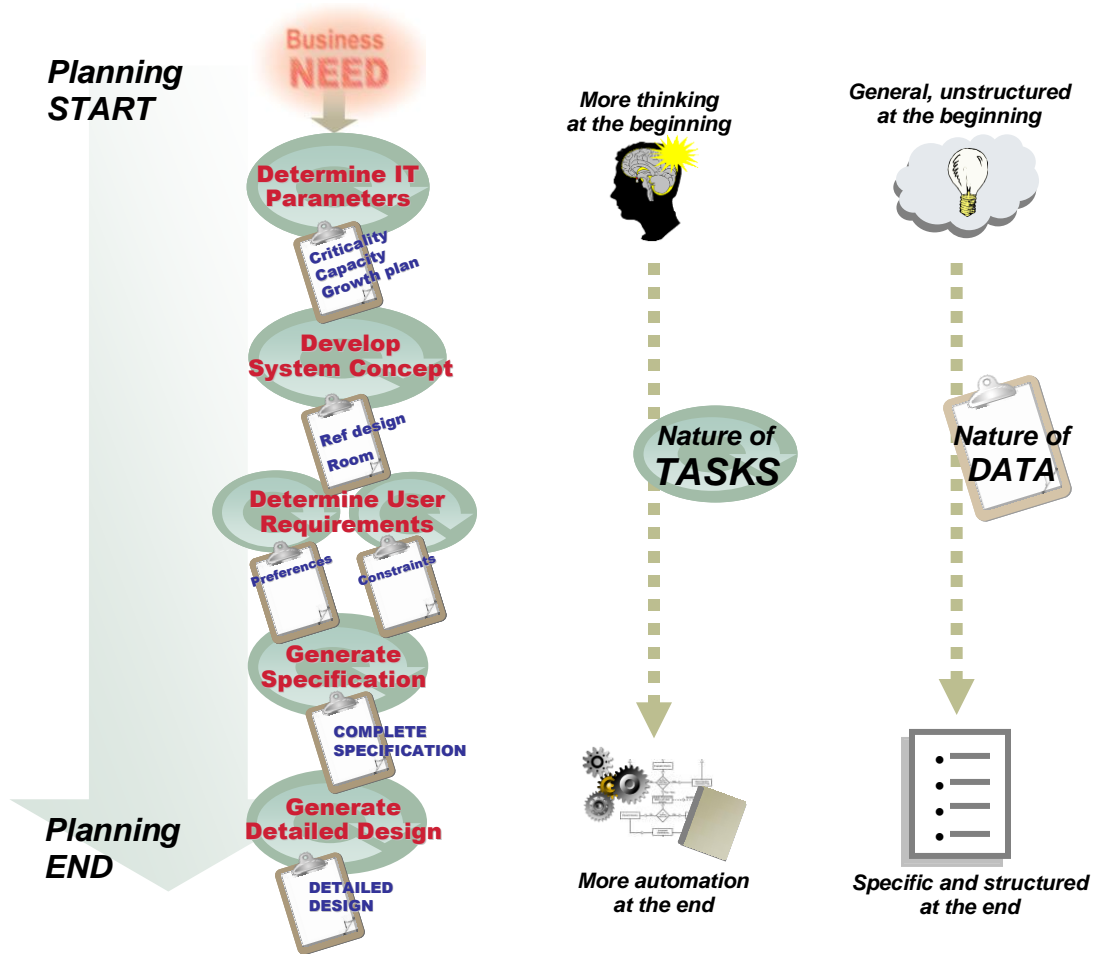


From abstract to specific

The output of each task becomes the input to the next task. At each task, input data is transformed by refining it, adding to it, or changing it to a different format, making it more specific to the system that is being designed. The early tasks rely primarily on human thinking to analyze, evaluate tradeoffs, make decisions, and come up with the required planning information. As the information becomes more structured, later tasks rely more heavily on automated methods, such as standardized specifications and software tools.

Figure 4 shows this progression from thought-based to automated, and general to specific.

Figure 4 – As the planning sequence goes from concept to final design, tasks become more automated and data becomes more structured



Hierarchy of Information

At each point in the planning process, the information becomes less abstract and more detailed. This hierarchy begins with the determination of three fundamental **IT parameters** that will directly affect the design of the physical infrastructure system:

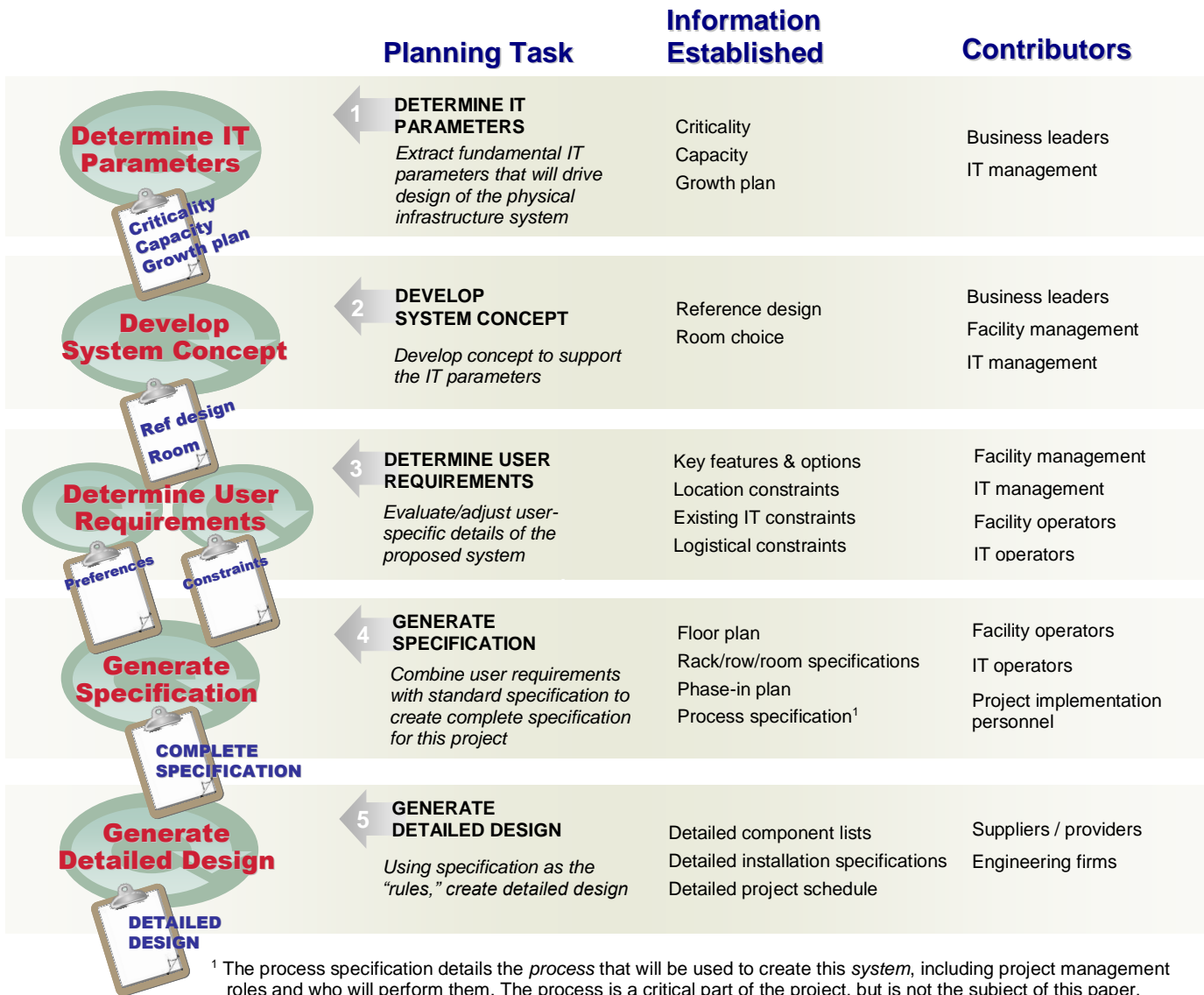
- **Criticality** – Business importance, in terms of tolerance for downtime
- **Capacity** – The IT power requirement (expected maximum after ramp-up)
- **Growth plan** – A description of the ramp-up to the maximum power requirement, incorporating uncertainty

Once these fundamental IT parameters are defined, a **system design concept** is determined. This can be facilitated by selecting one or more “reference designs” that are compatible with the three parameters, and compatible with the physical characteristics of the room that will be used for the installation. Next, specific details describing the user’s proposed system are collected and reviewed, to see what might need to be adjusted based on cost or other considerations. These user-specific details become the **user requirements**. The user requirements, combined with a standard data center specification, become the complete **specification** for the user’s data center. The specification serves as the “rules” that must be followed in creating the system’s **detailed design**. At the last level of the hierarchy, the **detailed design** is created. It includes specific products, layouts, and time schedules, all of which must comply with the specification. This final detailed design provides the blueprint for what will become the installed system.

As planning progresses from concept to design, from abstract to specific, different skills and expertise contribute to the process. At the earliest stages, business leaders supply the vision of the need that drives the project. Later on, decision-making shifts to individuals or groups who understand the technicalities and tradeoffs involved in how the physical infrastructure relates to other systems, including the IT system, existing electrical and mechanical facilities, and utilities.

Figure 5 shows the expected contributors at each level of the planning hierarchy, from business need to construction detail.

Figure 5 – Hierarchy of information in the system planning sequence



¹ The process specification details the *process* that will be used to create this system, including project management roles and who will perform them. The process is a critical part of the project, but is not the subject of this paper. The project process is described in APC White Paper #140, "Data Center Projects: Standardized Process."

Speaking at the right level of abstraction

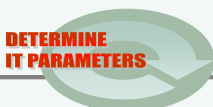
At any stage in the planning sequence, it is important to keep the discussion at the right level of abstraction – in other words, to speak only in terms of what is relevant to decision-making at that point in the process. Too little information, or information that is too abstract, doesn't provide enough guidance for decision-making. Too much information, or information that is too specific – or possibly even wrong, if outside the

expertise of the speaker – can overload key decision makers and unnecessarily constrain or mislead later tasks.

In the general-to-specific flow of information development, individuals who make decisions at each step must review and contribute to the plan at a level of abstraction that is appropriate to their knowledge and to their role in the process.

For example, in the earliest stage of the planning sequence (**Determine IT parameters**), it is *business leadership* who should be having the conversation, and they should be speaking in terms of business needs, not in terms of specific deployment techniques. Business leadership should be able to articulate a concept of the need as they see it, framed in their own language. At that point they should be able to exit the conversation, entrusting the design particulars to subsequent tasks that add actionable specifics to the concept, based on practical implementation considerations, requirements, and constraints.

The following are examples of statements that are appropriate to **Determine IT parameters**, and some that are not:

RIGHT: Appropriate level of abstraction <i>Focused on business need</i>		WRONG: Too specific <i>Focused on implementation details</i>
“We need a backup data center”		“The data center should be 500 watts per sq. ft.”
“Our criticality should be tier 3”		“We should use glycol heat rejection”
“Our lease may run out after 3 years”		“We should use K-rated transformers”
“We should be able to grow incrementally to accommodate uncertainty”		“We should use Brand X equipment”

Leadership directives that are too implementation-specific – and may not be fully technically informed – can misdirect subsequent downstream planning activity toward strategies that are unnecessary, expensive, or even impossible. Initial concept discussion should focus on statements that *directly reflect business need*. As the planning sequence progresses, more information is gleaned about the requirements of the particular installation; the data is refined into a progressively more specific description, ending with detailed what-goes-where installation instructions.

The sections that follow will examine this path of information flow by detailing each of the five steps along the way.

Task #1 in System Planning Sequence: Determine IT Parameters

This task starts with the general idea of a business need that requires a change to the organization's IT capability. From there, it makes a rough cut at determining three things that begin to quantify the plan for an improved (or new) IT capability. These three things are **criticality**, **capacity**, and **growth plan**. All are characteristics of the *IT function* of the data center, not of the physical infrastructure that will support it, which is the ultimate outcome of this planning sequence. In the task that follows this one, these IT parameters will be used to begin developing the physical infrastructure requirements for the data center.

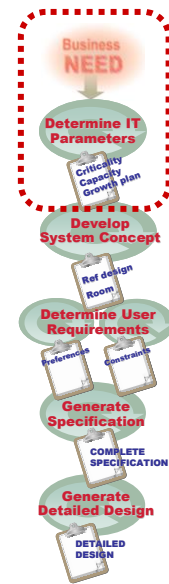


Table 2 – IT Parameters

IT Parameter	Expressed as ...	Description
Criticality	1, 2, 3, or 4 (4 is highest)	The goal for the availability and reliability of the data center, consistent with the business mission <i>For a summary of criticality levels, see -MGE White Paper #122, Guidelines for Specification of Data Center Criticality /Tier Levels</i>
Capacity	kW	Maximum IT power load during data center lifetime This number will be the “maximum final load” parameter in the IT load profile
Growth Plan	<ol style="list-style-type: none"> 1. Maximum final load (kW) 2. Minimum final load (kW) 3. Initial load (kW) 4. Ramp-up time (yrs) <p>IT Load Profile</p>	The expected IT load over the data center lifetime, expressed as the four-parameter IT load profile <i>For more about this growth model, see APC White Paper #143, “Data Center Projects: Growth Model”</i>

IT design comes first

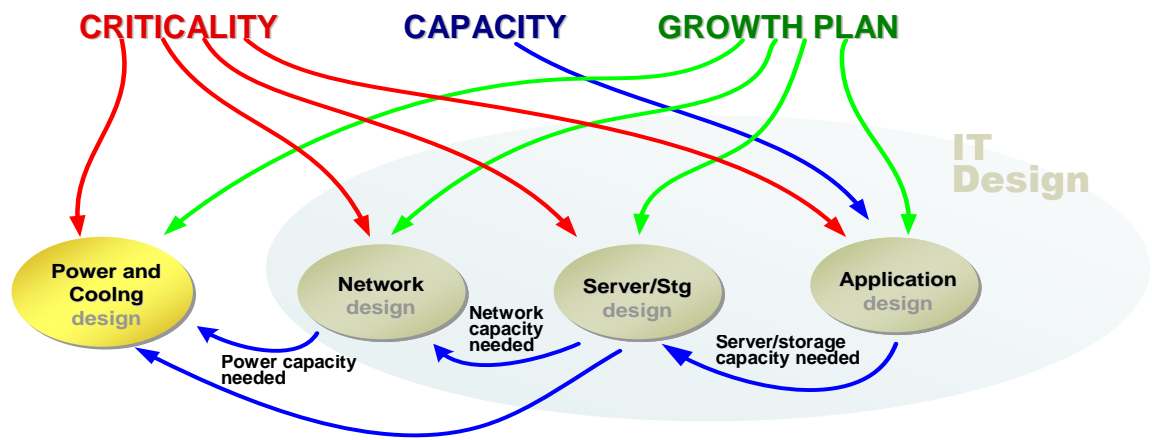
The planning sequence described in this paper covers the design of the data center's physical infrastructure layer, whose job it is to house, power, and protect the data center's computing and network functions – the IT layers. The beginning of this planning sequence assumes that the development of computing requirements and the design of the IT layers has already taken place. That discussion is outside, and precedes, the scope of this paper. However, the IT discussion generates essential thinking that becomes the foundational information for this planning process for design of the physical infrastructure system.

The IT discussion begins with computing needs, leading first to application design, followed by server and storage design, then network requirements. After that, the design effort shifts to the subject of this paper – the physical infrastructure system that provides power, cooling, management, physical security, and fire protection for the entire data center.

Figure 6 shows how criticality, capacity, and growth plan affect the chain of design that begins with computing requirements and progresses, layer by layer, to the physical infrastructure system. Note that criticality and growth plan directly affect the design of every layer. Capacity, however, begins with the concept of “how much computing” at the application layer, then flows logically to each succeeding layer as a support requirement that evolves from “how many servers needed” to “how much power needed.” From this it can be seen that criticality, capacity, and growth plan have already been considered when the time comes to design the physical infrastructure system.

The “IT parameters” – criticality, capacity, and growth plan – that begin the physical infrastructure planning sequence are merely a refinement of concepts that will have been addressed, to some degree, during IT design. In reality, however, there exists no standardized concept or language for these fundamental parameters of IT design, so they need to be clarified and quantified before they can be used to further guide the planning sequence described in this paper.

Figure 6 – The three “IT parameters” affect all layers of data center design



IT parameters: CRITICALITY

Criticality is a number from 1 to 4 representing how “important” the data center’s operation is to the business, in terms of toleration for downtime. Criticality is an expansion of the familiar concept of availability “tiers.” The selected criticality will determine the major characteristics of the system architecture, such as redundancy of power and cooling systems, as well as the robustness of system monitoring and various room construction details that affect reliability. **Table 3** provides a brief summary of criticality levels.

Table 3 – Summary of criticality levels

For a complete discussion of criticality levels, see APC White Paper #122, “Guidelines for Specification of Data Center Criticality / Tier Levels”

Criticality Level	Business Characteristics	Effect on System Design
1 (Lowest)	Typically small businesses; mostly cash-based; limited online presence; low dependence on IT; and perceive downtime as a tolerable inconvenience	Numerous single points of failure in all aspects of design; no generator; extremely vulnerable to inclement weather conditions; generally unable to sustain more than a 10 minute power outage.
2	Some amount of online revenue generation; multiple servers; phone system vital to business; dependent on email, some tolerance to scheduled downtime	Some redundancy in power and cooling systems; generator backup; able to sustain 24 hour power outage; minimal thought to site selection; vapor barrier; formal data room separate from other areas
3	World-wide presence; majority of revenue from online business; VoIP phone system; high dependence on IT; high cost of downtime; highly recognized brand	Two utility paths (active and passive); redundant power and cooling systems; redundant service providers; able to sustain 72-hour power outage; careful site selection planning; one-hour fire rating; allows for concurrent maintenance
4 (Highest)	Multi-million dollar business; majority of revenues from electronic transactions; business model entirely dependent on IT; extremely high cost of downtime	Two independent utility paths; 2N power and cooling systems; able to sustain 96 hour power outage; stringent site selection criteria; minimum two-hour fire rating; high level of physical security; 24/7 onsite maintenance staff

IT parameters: CAPACITY

This IT parameter answers the general question “How big a data center do I need?” **CAPACITY** (in this discussion) is the estimated maximum IT power load for the data center, over the data center lifetime.² It is *not* the power capacity of the physical infrastructure system to be designed by this planning sequence

² This CAPACITY parameter – meaning “maximum IT load” – will determine the upfront buildout of the *non-scalable* elements of the data center, such the service entrance and physical room size. It does not mean that all elements of the in-room physical infrastructure will be built upfront to support that power load, nor does it imply that the IT load will necessarily ever reach that level (it typically won’t).

(which will be higher) – rather, it is a best guess of the maximum IT load that will be supported during the data center lifetime. This number will become one of the four parameters in the **GROWTH PLAN**.

IT parameters: GROWTH PLAN

GROWTH PLAN is really four parameters – a set of four numbers that describe the expected growth of the IT power load, expressed in kW. These four numbers form the *IT load profile* that will guide the design of the power system. Uncertainty about future growth is handled in a simple way by providing both a maximum final load (the **CAPACITY** parameter, above) and minimum final load, and assuming the option of a scalable system design that approaches the maximum value in increments over time.

Figure 7a illustrates these four growth parameters. **Figure 7b** overlays a typical *system capacity plan* for the power and cooling system that will be designed to support the IT load profile. The system capacity plan is always designed to anticipate the maximum expected load, but the number and size of incremental deployment steps depends upon a number of factors, determined later in the planning sequence as the system design evolves.

For more about the this growth model, and how the system capacity plan is derived from the IT load profile, see APC White Paper #143, “Data Center Projects: Growth Model.”

Figure 8 illustrates the task detail for **Determine IT Parameters**.

Figure 7 – GROWTH PLAN is expressed as a 4-parameter “IT load profile”

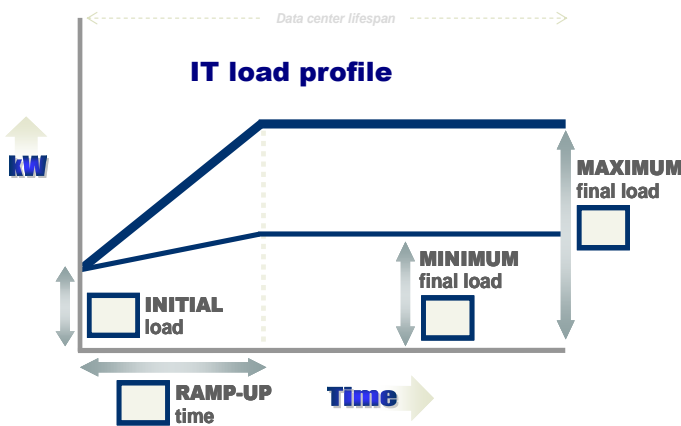


Figure 7a – The four parameters of the IT load profile



These describe the expected growth of the IT power load

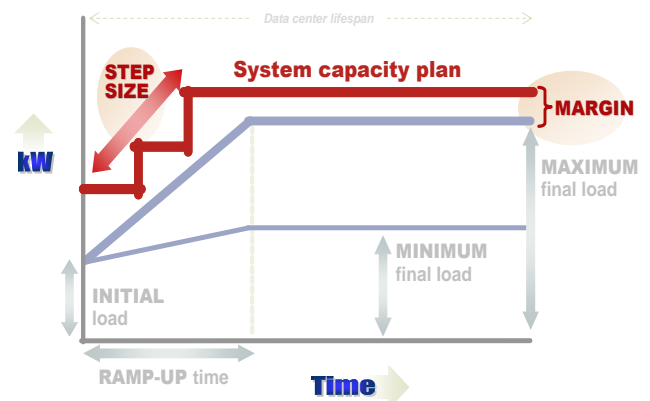
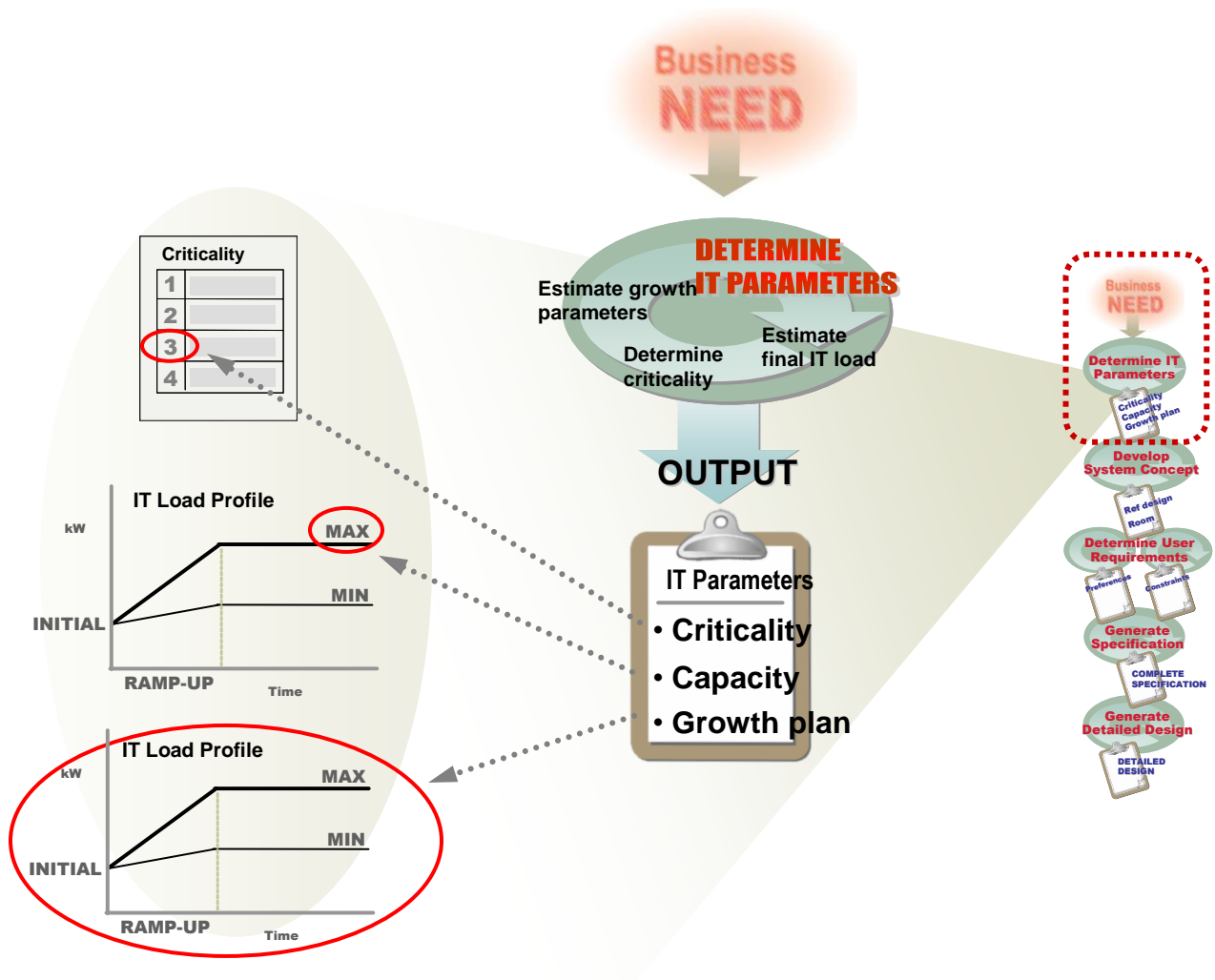


Figure 7b – The system capacity plan
Determined later, in task #4



*The **system capacity plan** is the planned deployment of power/cooling infrastructure to support the **IT load profile***

Figure 8 – Task detail for Determine IT Parameters



Task #2 in System Planning Sequence: Develop System Concept

This task takes the foundational IT parameters from the previous task – **criticality**, **capacity**, and **growth plan** – and uses them to formulate a general concept of the physical infrastructure system. The cornerstone of this task is the selection of a *reference design*, which embodies the desired **criticality** and **capacity**, and has a scalability that will support the **growth plan**.

What is a reference design?

Given the IT parameters of criticality, capacity, and growth plan, there are potentially thousands of ways the physical infrastructure system could be designed, but there is a much smaller number of “good” designs. A library of these optimal – i.e., recommended and proven –



designs can be used to quickly narrow down the possibilities. Much like a catalog of kitchen designs at a home improvement store, reference designs provide a choice of general architecture for the design of the system. Reference designs can help in zeroing in or ruling out, by presenting designs that may be hard to articulate or perhaps haven't been thought of. Reference designs have most of the system engineering built in, but with enough variability to satisfy the specific requirements of a range of user projects.

A reference design is an actual system design that is a prototype or "shorthand" representation of a collection of key attributes of a hypothetical user design. A reference design embodies a specific combination of attributes including criticality features, power density, scalability features, and instrumentation level. Reference designs have a practical range of power capacity for which they are suited.

The great power of reference designs is they provide a shortcut to effective evaluation of alternative designs without the time-consuming process of actual specification and design (tasks #4 and #5 in this planning sequence). High quality decisions can be made quickly and effectively.³

Choosing the physical room

Selection of the physical room where the system will be installed allows for consideration of room characteristics that might constrain the choice of reference design. Examples of possible constraints are room size, location of doors, location of support columns, floor strength, and ceiling height.

For example, suppose a reference design is not an exact physical fit in the user's room – racks may need to be added or removed. Such a reference design might have a cost attribute allowing estimation of the cost adjustment needed to scale the design to fit a specific floor space.

Choosing a reference design

The reference design is chosen to support the criticality, power capacity, and growth plan that have been determined in task #1 of the planning sequence. A reference design will have characteristics that make it more or less adaptable in each of these areas. For example, a reference design may be designed for one criticality level, or it may allow modifications to adjust the criticality up or down. A reference design may be able to support a range of power loads. A reference design may be scalable or not. Scalability of the reference design determines how well it can support the phase-in steps of a growth plan. Scalable reference designs may differ in the granularity (step size) of their scalability, making them more or less adaptable to a specific growth plan.

The IT load profile (constructed from the four parameters of the growth plan) will suggest a rough idea of phase-in steps. The reference design is then chosen to have a *scalability* that supports that general phase-

³ A reference design needn't come from a prepared library of designs, if there is another candidate that meets the desired criteria. A perfectly good reference design would be an existing data center identified by the user as a model for the one to be designed (although performance data may not be available, which would limit the ability to do TCO calculations).

in concept. Since the actual phase-in steps will be row-based, the row-by-row details of the phase-in plan will be finalized later in the planning sequence, after the row layout of the room has been determined.

A library of reference designs will be particularly useful if it has a software tool to assist in selecting an appropriate reference design. Input to such a “reference design selector” would be the foundational IT parameters established in the previous **Determine IT Parameters** task – criticality, capacity, and growth plan. Other essential requirements may also need to be included to further narrow down the possible reference designs, such as type of cooling or power density. The automatically selected reference design(s) can then be reviewed for additional considerations that may not be handled by the selector tool, such as location of doorways, support columns, or other significant constraints.

Using reference designs to develop the system concept

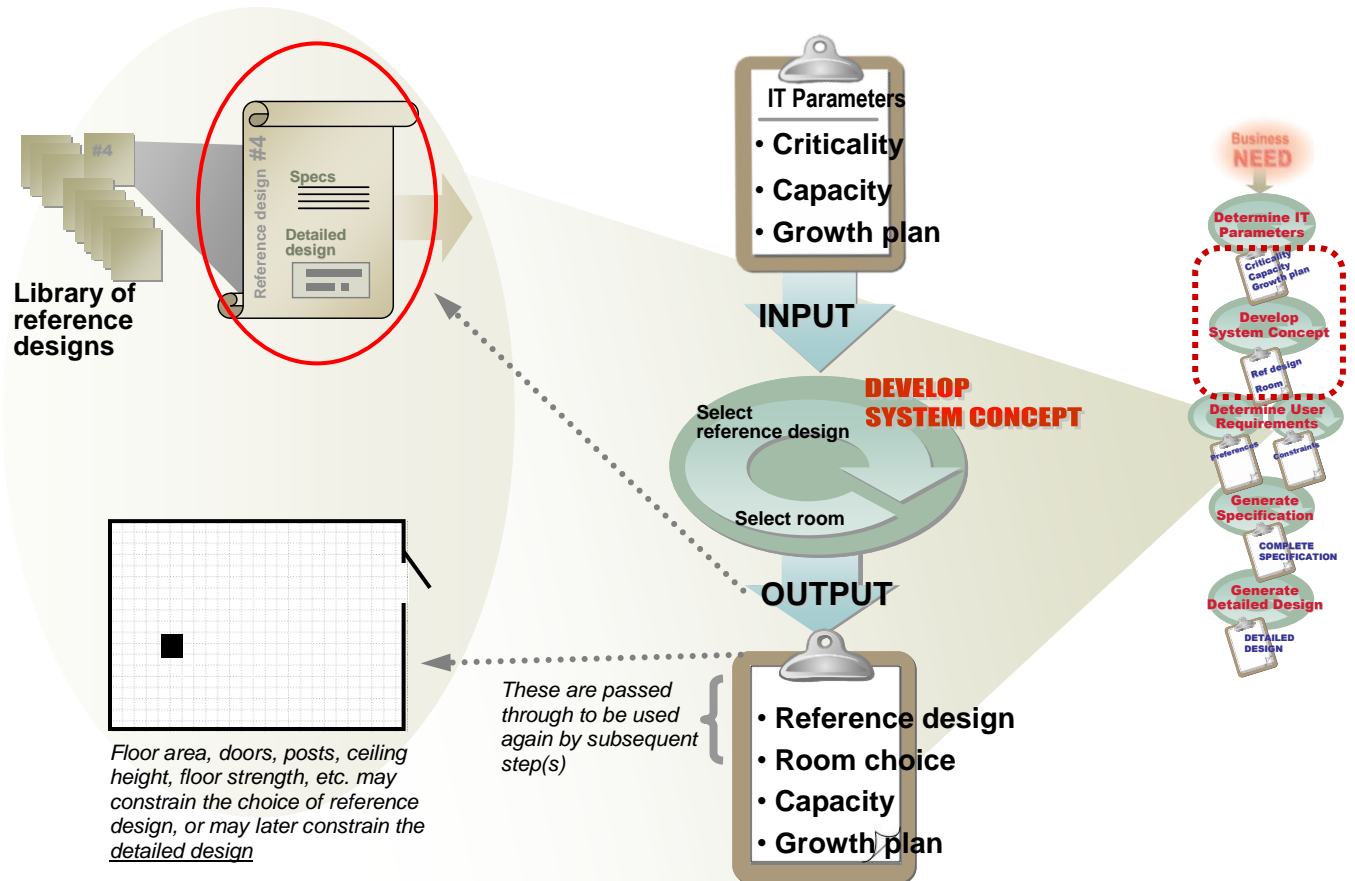
In a library of reference designs, there may be more than one design that is compatible with the IT parameters and other requirements specified by the user. The user may want to compare these reference designs to each other or to other alternatives, such as reference designs from other vendors or even to outsourced custom-engineered designs. The ability to perform such comparisons is a valuable asset in developing the physical infrastructure concept. Essential attributes for comparison include

- TCO
- First cost
- Electrical usage
- Density capability
- IT equipment capacity (for a given floor space)

Calculation tools that take the user’s IT parameters and combine them with a given reference design to estimate the above information can assist in making informed decisions regarding selection of a reference design. The most useful reference designs will therefore be documented with the specific values of their essential characteristics – for example, efficiency, density, cost, and other parameters – which can be used to calculate the above information.

Figure 9 illustrates the task detail for **Develop System Concept**

Figure 9 – Task detail for *Develop System Concept*



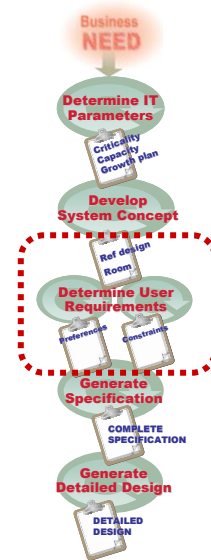
Task #3 in System Planning Sequence: Determine User Requirements

User requirements include any information about the project that is specific to *this* user's project. This task collects and evaluates user requirements to determine whether they are valid, or should be adjusted in some way to reduce cost or avoid problems. *User requirements* for the project can be such things as key features and options, room constraints, existing IT constraints, and logistical constraints. This task has two halves, dividing user requirements into two general categories:

Preferences – Things that the user would like, but are subject to change or adjustment after consideration (or reconsideration) of cost and consequences. *Preferences are things you want, but can change your mind about if you get new information.*

Constraints – Things that either cannot be changed, or can only be changed at great expense or with unacceptable consequences. *Constraints are pre-existing conditions that are difficult or impossible to change.*

Each requires a different type of discussion, often carried out with different types of people. If something causes trouble later in the planning process, it is important to know whether it was a constraint or a preference, so it can be dealt with correctly and adjusted if appropriate.



Earlier preferences and constraints

Some preferences and constraints have already been identified earlier in the planning sequence, during selection of the reference design in the preceding task (**Develop System Concept**). For example, target capacity is a *preference* (since the future is uncertain), and the size of the physical room is a *constraint*. Preferences and constraints that are needed for selection of the reference design can be collected in advance or can be established in an interview process during reference design selection. They should then be added to the preferences and constraints being catalogued in *this* task, because they may affect other elements of the design.

Preferences

Preferences are design decisions made by the user based on experience, convenience, or business judgment. A preference can be re-evaluated and changed upon consideration of cost and consequences. Preferences can be related to the system (“I want overhead power cabling”) or the process (“I want to use Ace Electric to do the electrical work”). Preferences regarding specific technical details should generally be avoided at this point in the process, since the technical specification and design of the system don’t occur until later in the process, and may conflict with technically uninformed preferences expressed here (see earlier section under task #1, **Speaking at the right level of abstraction.**)

In general, preferences are

- Something you *want*, rather than something you are stuck with
- Something you'd like to do for business reasons
- Something you might change your mind about, if you discover new information
- Something you can decide to change later in the process, if new information develops

Note that the clipboard items **growth plan**, **capacity**, and **reference design** all flow into the PREFERENCES half of this task. Except in very rare cases, these are all preferences, not constraints.

Constraints

Constraints are dictated by circumstances and not under the control of the customer. A constraint is something you didn't ask for, but rather exists on its own, independent of business needs. Constraints include facility limitations, regulatory limitations, or unchangeable business requirements.

Changing a constraint would be expensive, troublesome, or impossible. Constraints are usually not a subject for debate – for example, the existence of a post in the middle of the room. However, some constraints can cause so much trouble downstream that a decision is made to fix them, because it is deemed worth the cost.

Examples of constraints:

- A physical characteristic of the room (such as ceiling height)
- A law or code that requires compliance
- A standard that you are committed to meeting (such as TIA 942)
- A physical characteristic of the delivery path (such as the weight capacity of the elevator that will be used to transport equipment to the room)

Preference or Constraint?

It is important to know whether something is a preference or constraint, especially if it causes trouble later in the planning process. For example, target capacity in most cases is a preference, not a constraint, since the future is uncertain. Suppose the original IT parameters call for a one megawatt (MW) data center, expandable to 4MW. Further along in the process, an engineer knows that the utility feed can only provide 3.5MW. Assuming that the 4MW expansion plan is immutable, the engineer plans to carry the expansion to 3.5MW, then spend a large amount of money to upgrade the utility feed for the last 500kW. Interpreting the 4MW as a “constraint” cast in stone prevents the engineer from going back up the chain of planning to review the decision. If the 4MW target capacity had been recognized as a “preference” – subject to change with new information – the engineer would go back to discuss the situation and review alternatives. This allows the original planner to reconsider and say “Oh, I didn't realize that – let's make it 3.5MW instead of 4MW.”

It is important to make sure that a constraint really *is* a constraint. Here are two examples of a work-around for a constraint:

Constraint: The data center is not allowed to be turned off (to perform the upgrade).

Possible work-around: Put up a temporary wall to separate the running system from the work area for the new installation, and bring in a separate utility feed to run both systems at once during changeover.

Constraint: We can't use air-removal units that duct into the dropped ceiling, because the ceiling has a non-fire-rated plenum duct and the fire inspector won't allow a hookup to it.

Possible work-around: Spray the duct with fireproofing.

Note that **Room choice** (one of the four input items to this task) flows into the CONSTRAINTS half of the task. This is because the room is an existing physical thing that may have characteristics that become constraints to the system design.

Evaluating and refining preferences and constraints

Some preferences and constraints can have a significant negative impact on the project. Each *preference* must be examined to determine whether there are any costs or consequences that might prevent it from being a good business value. For example, if the data center is intended to be a showcase, a glass wall might be a preference. However, the financial cost and security risk might rule it out. Each *constraint* also must be examined to determine its effect on the system design, and whether it is worth the financial cost or other negative consequence of removing it.

In both cases, alternatives are identified, tradeoffs are assessed, and the preference or constraint is kept, adjusted, or eliminated as appropriate.

For complex projects, some preferences and constraints may have an exceptionally serious negative impact. Expert review is essential in order to establish their cost or other consequences, which may not have been obvious to the initial planners. Once the consequences of certain preferences and constraints are clear, it is important to review them and determine if they can be refined or adjusted to achieve a better overall result.

What happens to preferences and constraints?

Some preferences and constraints may have already been used in the preceding task (**Develop System Concept**) to select a reference design. All are passed on to the next task (**Generate Specification**) where they become the user-specific portion of the complete project specification.

The vast majority of preferences and constraints are not needed to select the reference design (a previous task), but *are* needed to create the complete specification and the detailed design (subsequent tasks).

These preferences and constraints can be categorized into the following areas:

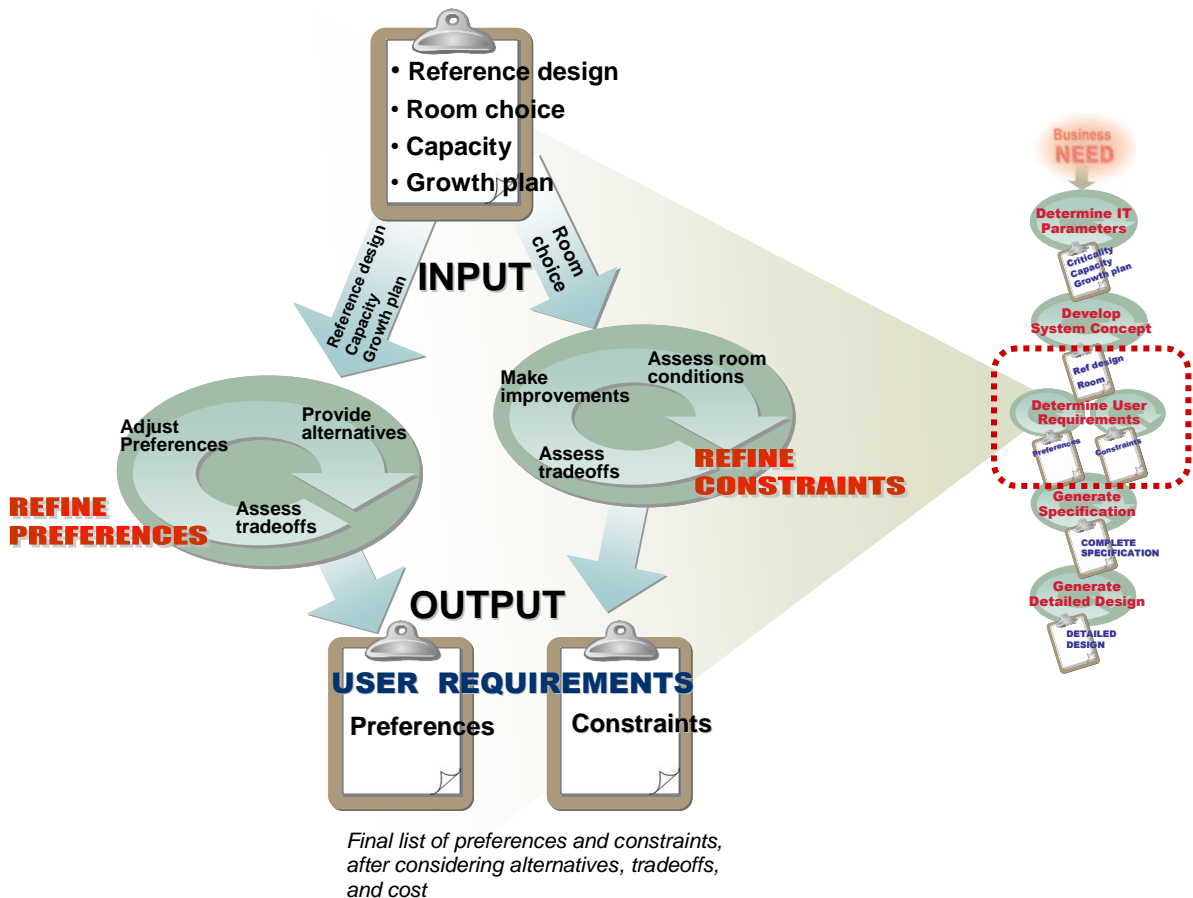
- Schedule
- Division of work
- Facility electrical constraints
- Facility cooling constraints
- Structural room constraints
- Re-use of existing infrastructure
- Phase-in plan

Note that these preferences and constraints are in addition to original IT parameters **criticality**, **capacity**, and **growth plan**.

Once the preferences and constraints are established, a specification for the system can be created (next task), because the preferences and constraints become the **user specification** portion of the complete system specification. For more about preferences and constraints, see the next task (**Generate Specification**), where they are described in more detail under their new name, **User Specifications**.

Figure 10 illustrates the task detail for **Determine User Requirements**.

Figure 10 – Task detail of **Determine User Requirements**



Task #4 in System Planning Sequence: Generate Specification

To translate the user requirements (from the previous task) into a detailed design (in the task following this one), this planning sequence uses the intermediate step of creating a system **specification**.

The system specification serves as a set of rules to be followed when creating the detailed system design. The specification consists of the following elements:

Standard specifications that do not vary from project to project. These standard specifications comprise the major portion of the specification. Examples of standard specifications are regulatory compliance, compatibility of subsystems, workmanship, safety, and best practices.

User specifications that define the user-specific details of *this* project. These are the **user requirements** (preferences and constraints) from the previous task, which come from the accumulated work of the planning sequence up to this point.⁴

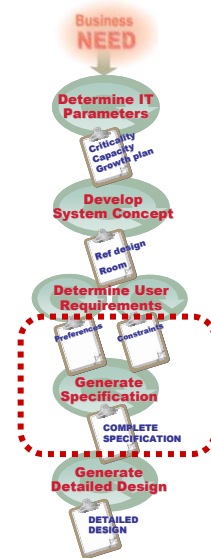
The combination of the standard specifications and the user specifications creates a complete specification that serves as a “rulebook” for the detailed system design. The detailed system design (created in the next, and final, task) must meet all the specifications in the complete specification.

Separating the specifications into those that are common to *all* systems (the standard specification) and those are particular to *this* user’s system (the user specification) greatly simplifies the job of specifying the complete system, because the body of standard specifications is much larger than the user specifications. The user-specific specifications can be filled in on worksheets and attached to the standard specification, together forming the complete system specification for the project.

User-specific elements of the specification

The user-specific portion of the system specification includes information that describes, in detail, everything that is particular to this user’s project. This includes information about all elements of the physical infrastructure system itself, as well as information about other structures and systems that affect the design of the physical infrastructure. The following is a sampling of the user-specific information included:

- Physical characteristics of the room
- Electrical service entrance



⁴ The user specification also includes user-specific details of the *process* that will be used to create this *system*, including project management roles and who will perform them. The process is a critical part of the project, but is not the subject of this paper. The project process is described in APC White Paper #140, “Data Center Projects: Standardized Process.”

- Row-based peak and average power
- UPS runtime required
- Power distribution characteristics
- Phase-in plan
- Type of heat rejection
- Type of fire suppression
- Type of physical security
- Existence of building and/or network management system

There are three levels of abstraction for the user-specific information: **room**, **row**, and **rack**. The flow of specification (filling out the worksheets) begins at the room and moves to the more specific row or rack level, as needed.

Floor plan

The floor plan is a key part of the user specification. Floor plan is a specification issue, not a design issue, because the floor plan is needed in order to specify density variation and phase-in plans, which are an essential part of the overall specification.

The floor plan must show the location of rows, but does not need to specify whether individual rack positions within each row contain IT or physical infrastructure equipment.

During creation of the detailed design (the task following this one), some rack locations will be consumed for physical infrastructure devices. An *estimate* of IT rack space at the specification stage (this task), can be made based on simple rules, since the basic architecture of the physical infrastructure is known at this point, but the *exact* IT space available will not be determined until the detailed design is completed.

If the design includes devices that are not rack-compliant, such as wall-mount CRACs, space must be reserved for them on the floor plan.

The principal design element in the floor plan is the **row**. Rows are any group of one or more contiguous rack locations. A long row with a gap in the middle is considered two rows. Row locations are laid out using the methodology described in APC White Paper #144, "Data Center Projects: Establishing a Floor Plan."

Room-level specifications

Some specifications always apply at the room level, such as physical security, fire suppression, and utility power requirement. Some projects may also have existing conditions (such as a raised floor) that must be included in the user specification; these are almost always at the room level.

For Readers Who Are APC Customers

It is not necessary to specify the location of separate physical infrastructure devices in the floor plan (such as CRACs around the room perimeter). This is because the devices will either consume no floor space (mounted in racks) or are themselves rack-based and become part of the row layout. The device selection and location are considered to be part of the design (next task), not part of the specification (this task).

Some types of specifications can be specified at the room level, or optionally at the row or rack level. Examples of specifications that can apply to individual rows or racks are criticality, power density, rack type, and deployment phase. When these specifications are uniform across the data center, they are best treated as a “room” specification.

Row-level specifications

A floor plan includes one or more rows, which can be of any length. Some attributes can be very effectively specified at the row level. Examples are criticality, power density, and deployment phase (described later).

NOTE: The design methodology used by APC (next task, **Generate Detailed Design**) is optimized for row-level specification. It is a best practice for rows to be defined as the unit of criticality, density, or deployment phase, with uniform specifications within each row.

Rack-level specifications

The ideal of row-level specification is not always required or desirable – for example, when unique and specific racks such as SAN devices or wiring patch panels are known in advance and are installed in a row. In these cases, the specification allows for rack-level specification on an exception basis, or for every rack in the row. However, rack-level specification for every rack within a row can over-constrain the specification so that necessary power and cooling devices cannot be located within the row. Therefore, rack-level specification should be used only on an exception basis, and row-level specification should control most of the rack characteristics.

It is not only possible, but expected, that some specifications will include NO rack-level specifications.

The phase-in plan

Given a floor plan with rows identified, a deployment phase-in plan can be created. The phase-in plan (part of the **User specification**) describes the step size and timing of incremental deployment. The phase-in plan is developed from the IT load profile that was determined earlier in task #1, **Determine IT Parameters**” as shown in **Figure 11**.

The step size is determined based on both strategic and tactical considerations:

Strategic. The general concept of step size – few steps, many steps, or no steps – is determined based on ramp-up time, difference between initial and maximum load, and the uncertainty regarding final load. These issues are discussed in detail in APC White Paper #143, “Data Center Projects: Growth Model.” This growth model takes into account timing and uncertainty.

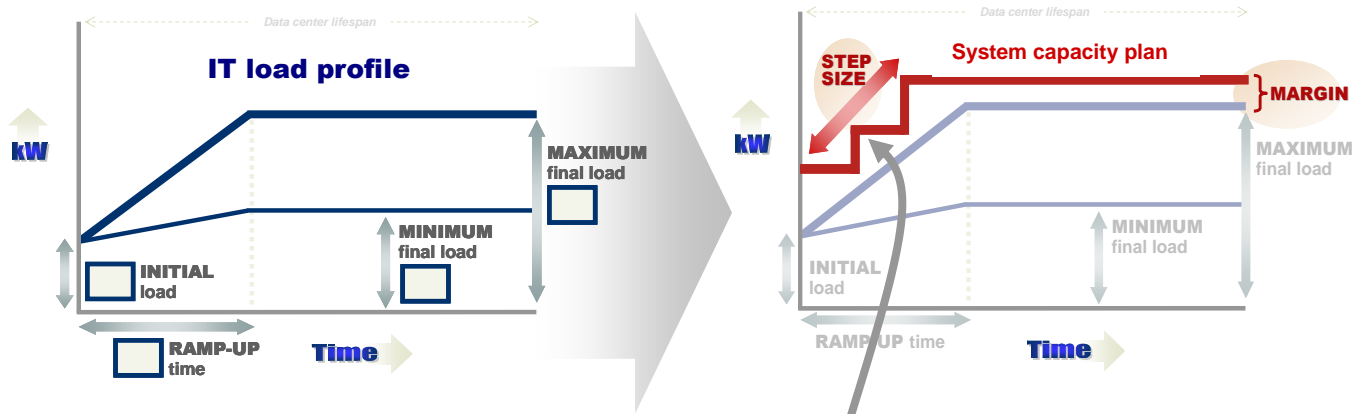
Tactical. The specific, hardware-based step size is determined from the physical configuration of rows in the floor plan. Best practice step size is row-based, making each step a row or a group of rows. (This assumes that physical infrastructure support is also row-based, and each step will

deploy an integrated zone with both the IT equipment and the physical infrastructure equipment required to support it.)

Future phases often have significant uncertainty regarding their timing, power density, and even whether they will ever occur. Projects with long-range phase-in of high uncertainty can be specified with highly modular architecture, to maximize electrical efficiency and avoid stranded capacity investments. When specifying density for future phases it is best to be safe and over-specify as practical. This means that key supporting infrastructure such as water supply piping and electrical runs are installed up front to support the maximum potential load. This does not necessarily mean that the full projected buildout of CRACs, UPSs, and PDUs will actually occur. In a typical design, it is possible to downgrade the density specification prior to a future deployment, but it can be very difficult to upgrade the density specification if key supporting infrastructure is missing.

Figure 11 – The phase-in plan is developed from the IT load profile

For more about this growth model, see APC White Paper #143, "Data Center Projects: Growth Model"



The IT load profile is determined in the first task, **Determine IT Parameters**

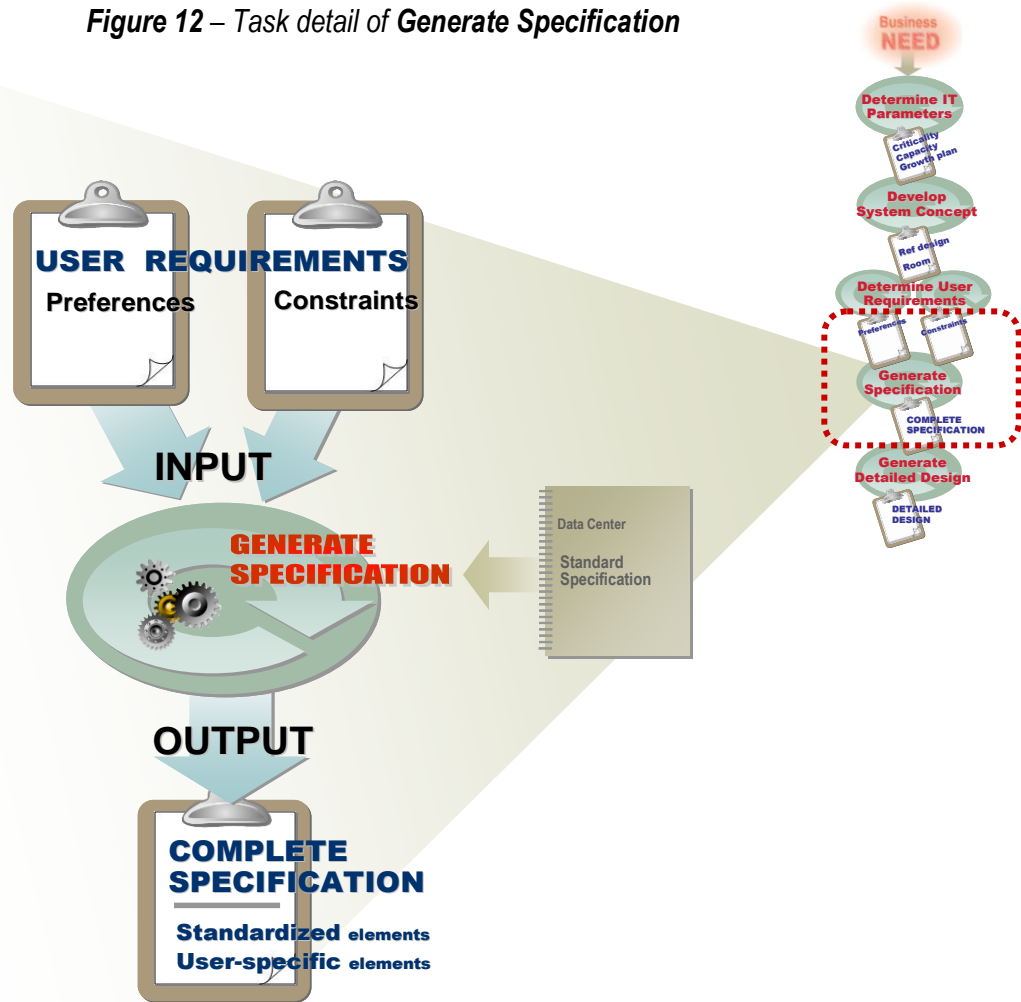
The phase-in plan consists of the size and timing of steps during the ramp-up to full capacity

Process specification

A project consists of both the *system* that is created and the *process* that does the work to plan and build it. The subject of this paper is the planning of the *system*, but there is also a parallel effort to configure the steps, work assignments, and management roles of the *process*. The specification for the total project therefore includes a specification of the process, including both standardized and user-specific elements of the process. For more about this process, see APC White Paper #140, "Data Center Projects: Standardized Process."

Figure 12 illustrates the task detail for **Generate Specification**.

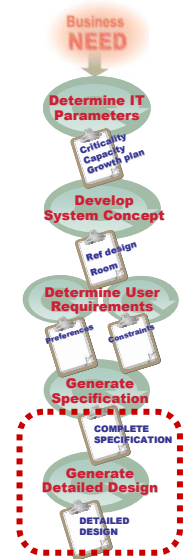
Figure 12 – Task detail of Generate Specification



Task #5 in System Planning Sequence: Generate Detailed Design

The last task in the planning sequence is the creation of a detailed design for the installed system, including

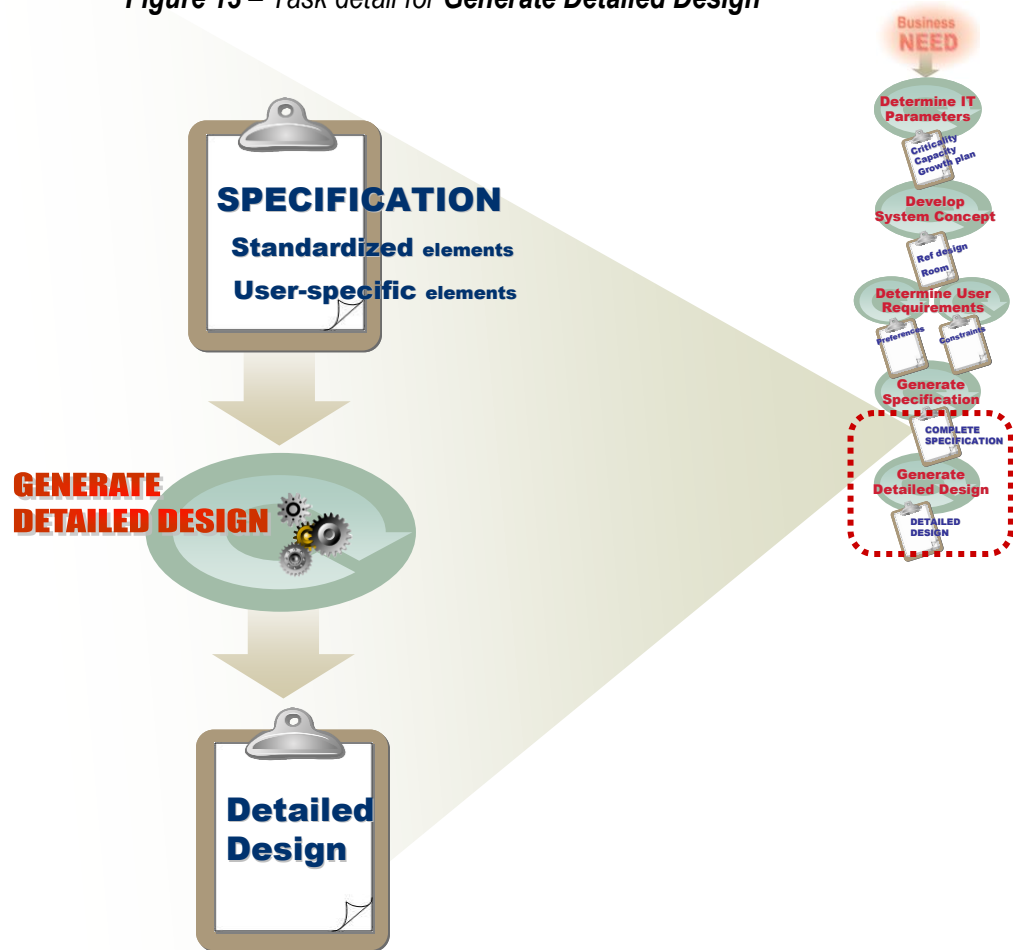
- Detailed component lists
- Exact floor plan of racks, including power and cooling equipment
- Detailed installation instructions
- Detailed project schedule
- Actual “as built” characteristics of the design, including efficiency, density, and expandability



The intent is that the detailed design will meet the complete specification created in the previous task. The complete specification includes specifications for the user-specific details of *this* system, plus an extensive body of performance-based specifications that apply to *any* system. To the extent that it meets all those “rules,” the detailed design will represent the system described by the specification.

Ideally, if the complete specification is robust and clear enough, the detailed design *could* be automatically generated from it. However, that is not typically the case. Creation of the detailed design, even if partially automated by a design tool (as it is at APC), will require the involvement of a professional engineer, much like an architectural drawing needs a person even if the building has already been fully architected. If the planning sequence has been carefully executed according to the model described in this paper, there should be no need for any significant decisions at this point. The only requirement is someone who has the tools and expertise to correctly produce the output (the detailed design) from the input (the specification). **Figure 13** illustrates the task detail for **Generate Detailed Design**.

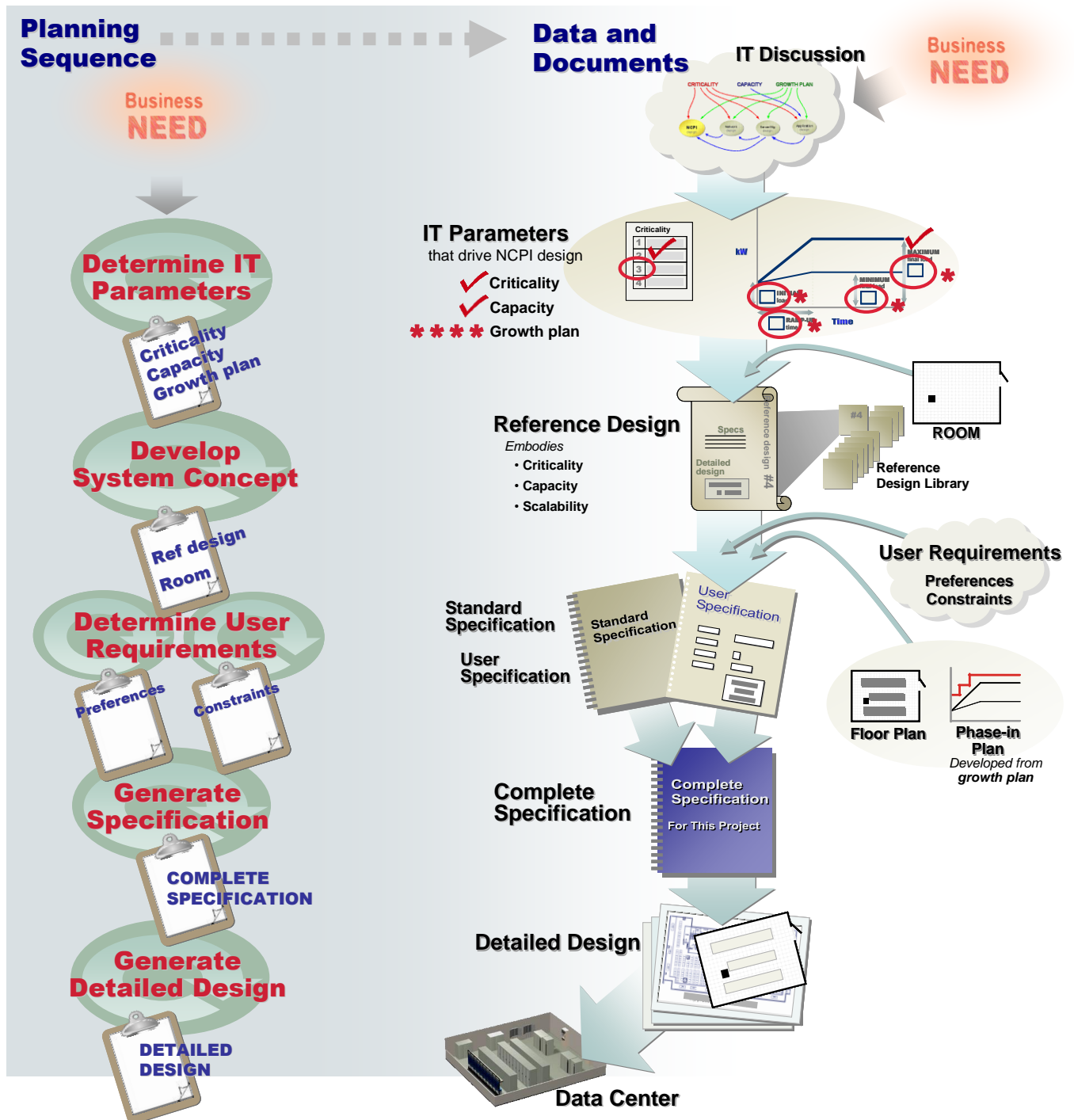
Figure 13 – Task detail for Generate Detailed Design



Summary of Data and Document Relationships

During the course of the planning sequence, a variety of documents and data sets interact and evolve. **Figure 14** maps the interactions and dependences among these various items of information. The data-flow model is reproduced beside it for reference.

Figure 14 – Summary of planning sequence and corresponding data and documents



Conclusion

Despite its crucial importance to the success of the project, system planning has historically been considered unstructured and difficult, carried out as art rather than science, with opportunities for missteps, wrong assumptions, and miscommunication that can have serious consequences in later phases of the project. Much of the difficulty can be removed by viewing system planning as a standardized data-flow model, consisting of an orderly sequence of tasks that progressively develop and refine the system concept, to ensure that the final system fulfills the original business need.

In the data center project process, it is the two planning phases – **Prepare** and **Design** – that lay the essential foundation for the project. In addition to administrative and process tasks, threaded through these two phases is a sequence of tasks that accomplish the planning of the physical system. Each of the tasks in this “system planning sequence” is a subprocess that refines or transforms the system concept as it progresses from idea to detailed design. The sequence starts with

A **Business Need**, from which is distilled
 IT Parameters, from which is developed a
 System Concept, detailed with
 User Requirements, refined and integrated with standards to become
 Specifications, which are the “rules” that must be followed by the
 Detailed Design

At each of the levels in this hierarchy, the information becomes less abstract and more detailed. The tasks are defined in a way that provides the right amount of information, at the right level of detail, at the right time – the necessary and *sufficient* information at every point. In this way, re-work is reduced, cycle time is accelerated, and individuals who make decisions in the process will be able to review, and if necessary approve, the information at the level that is appropriate to their role. Informed and expert attention to the execution of this planning sequence is a critical element in achieving efficient process flow and a successful project outcome.

Data center planning has matured to the point where it no longer requires heroic artisanship, as it did earlier in its decades-long history. It is time to consolidate this storehouse of knowledge into a standardized process, so creative engineering skill can focus on unusual, complex, and very large installations that require specialized or exceptional expertise. Standardization, a common language, and adherence to a well-defined process can bring most data center planning into the realm of predictable, repeatable science.

About the Author

Suzanne Niles is a Senior Research Analyst with the APC Data Center Science Center, where she develops white papers and presentations on technical and strategic topics that support the APC mission. She studied mathematics at Wellesley College before receiving her Bachelor's degree in computer science from MIT, with a thesis on handwritten character recognition.

From 1971 to 1981 Suzanne worked on the development team that created Express, a pioneering multidimensional data management system (now part of Oracle). She has been educating diverse audiences for over 30 years using a variety of media from software manuals to photography and children's songs.

Related White Papers

APC white papers about the data center project process

<i>White Paper</i>	<i>Subject</i>
#140	Standardized Process
#141	Project Management
#142	System Planning <i>[this paper]</i>
#143	Growth Model
#144	Establishing a Floor Plan