It has been observed that "The life cycle is the only thing that uniquely distinguishes projects from non-projects"[1].

This is because a project or programme life cycle illustrates the distinct phases that take an initial idea; capture stakeholder objectives; develop a set of requirements and then deliver the objectives. There is a beginning and an end, as distinct from a continuous operation.

The goals of life cycle management are to:

- identify the type of life cycle that best matches the context of the work;
- define the phases and gates that will form the life cycle;
- structure governance activities in accordance with the life cycle.

Projects and programmes are the primary mechanisms for delivering objectives while portfolios are focused on co-ordinating and governing delivery of multiple projects and/or programmes. As a result, the project and programme life cycles have many similarities and follow the same basic approach.

Since projects and programmes do come to an end, there is an argument to say that they do not have a true 'life cycle' but rather have a 'life span'.

Portfolios cycle perpetually through repeated phases and so are much closer to the real concept of a life cycle.

Since 'life cycle' is by far the most recognised term, that is the one used throughout Praxis for projects, programmes and portfolios.

Project and programme life cycles

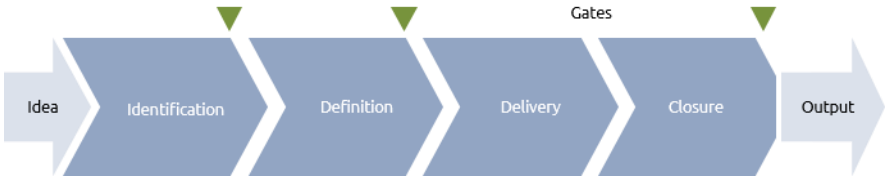
There are broadly four types of project and programme life cycle. Determining which type is most appropriate in any given context is based primarily on the nature of the objectives and the way in which they should be delivered.

Type	Objectives	Delivery
Linear	Stable and predictable	Single delivery at closure
Incremental	Reasonably stable but can more easily accommodate change	Staged delivery resulting in earlier benefits and return on investment
Iterative	Dynamic because of uncertainty	Single delivery at closure
Incremental and iterative	Dynamic because of uncertainty	Staged delivery resulting in earlier benefits and return on investment

Many projects and most programmes will use combinations of these approaches because different elements of the work will have different objective and delivery characteristics.

The following explanations of the four types of life cycle should not be seen as describing four types of project or programme. Rather, they are four ingredients from which a context specific life cycle can be formed.

Linear

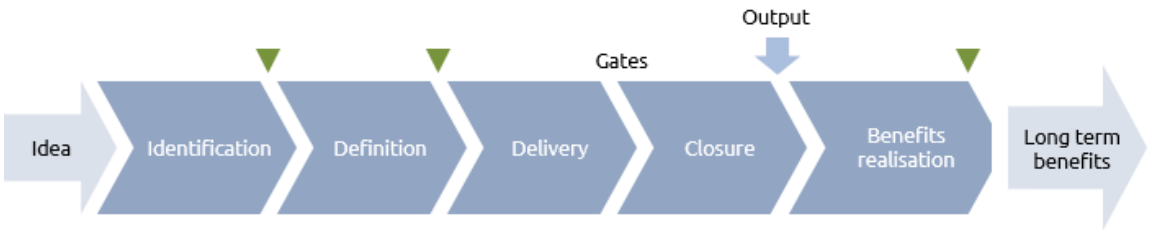


It all starts with someone having an idea that is worth investigation. This triggers high level requirements management and assessment of the viability of the idea to create an outline business case. At the end of the identification phase there is a gate where a decision is made whether or not to proceed to more detailed (and therefore costly) definition of the work.

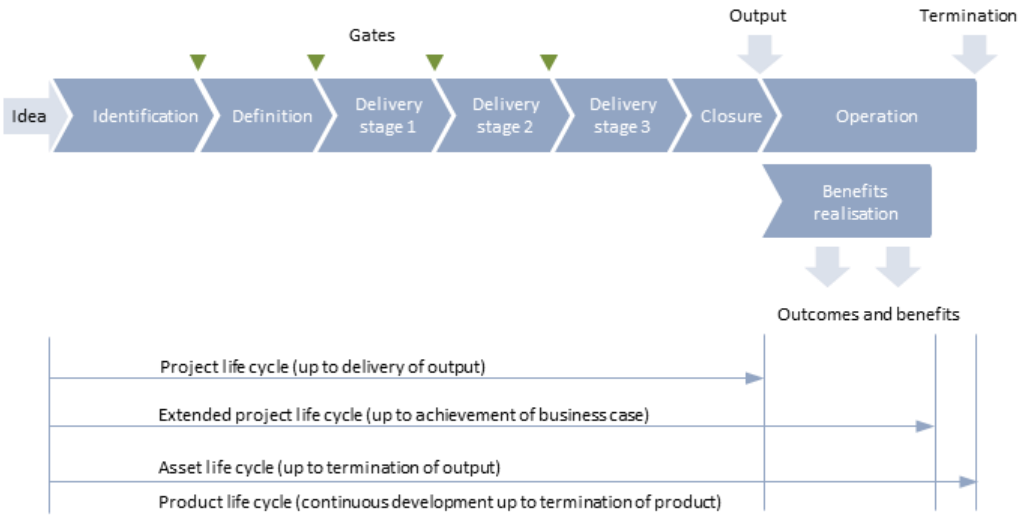
If the idea is good enough, the work will continue to a detailed definition that produces a full justification for the work. Once again, this ends in a gate where a decision is made whether or not to proceed to the delivery phase.

Once the output has been produced it is usually subject to an acceptance process before being formally delivered to its new owner. The life cycle comes to an end with the closure of the project.

All outputs are intended to deliver benefits and this can be shown as an additional phase:



Ultimately, the outputs from which benefits are derived will reach the end of their natural life and be decommissioned or terminated. Life cycles that extend all the way through to the termination of the outputs are typically called ‘asset life cycles’ or ‘product life cycles’.



- Identification – in this phase the initial idea is developed and a project brief is created. A sponsor is appointed and, if possible, a project manager. Sufficient analysis must be performed to enable senior stakeholders, led by the project sponsor, to make two decisions:
    - Is the project likely to be desirable, achievable and viable?
    - Is it worth investing in the definition phase?
- Definition – in this phase the requirements are assessed in greater detail and the preferred solution specified. The management plans, delivery plans and business case are developed and these have to be approved by the sponsor before progressing to the next phase.
- Delivery – this phase may be further broken down into stages. At the end of each stage the continuing justification for the project is be reviewed.
- Handover and closure – the project outputs are handed over and accepted by the sponsor, client or users as required.
- Benefits realisation – where appropriate, a project may include a benefits realisation phase. This is typically done where there is a non-complex, one-to-one relationship between an output and the benefit (more complex relationships between multiple outputs and benefits are governed as a programme).

The asset life cycle also includes:

- Operation – continuing support and maintenance;
- Termination – closure at the end of the product's useful life.

Another popular term (particularly in the software industry) is the product life cycle. This is very similar to the asset life cycle but involves continuous development of the product alongside its operation.

These basic life cycles can be adjusted to many different contexts, e.g. in circumstances where the work is:

- performed by a contractor on behalf of a client;
- a project that is part of a programme;
- specific to a particular industry or profession.

Understanding life cycles can be helped by looking at them in three ways:

Governance life cycles are generic approaches to structuring the way an initiative is managed.

Specialist life cycles are governance life cycles that address a specific business context.

Development life cycles are a sub-set of governance life cycles that focus on product development.

The phased structure of life cycles facilitates the creation of governance mechanisms, such as:

- Defined processes – the management of each phase can be described as a process made up of a number of relevant activities.
- Stages and tranches – the delivery phase can be subdivided into packages of work, typically called stages on projects and tranches on programmes.
- Gate reviews – these are conducted at the end of a phase, stage or tranche. The sponsor will consider performance to date and plans for the next phase, stage or tranche before deciding whether the business case remains viable, practical and achievable.
- Post-reviews – learning from experience is a key factor in maturity. Post-project and post-programme reviews document lessons learned for use in the future.
- Benefit reviews – these measure the achievement of benefits against the business case.

This type of life cycle depends upon the ability to substantially specify the objectives of the work, and how they will be achieved, by the end of the definition phase. This does not mean that changes cannot be accommodated during delivery.

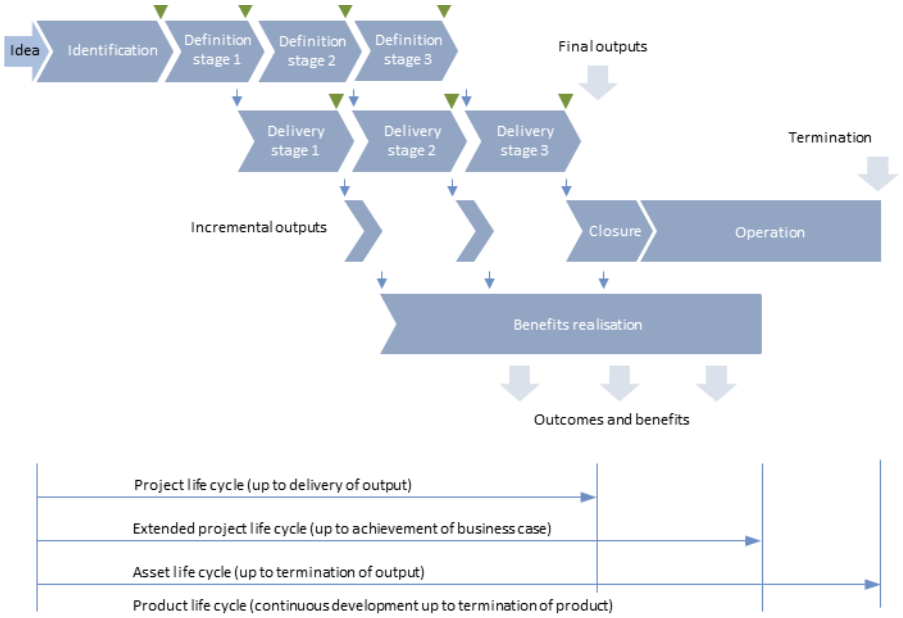
Change control is applied during delivery but the cost of change will affect the viability of change requests and the work completed so far will affect the practicality of proposed changes, i.e. the further the work progresses, the harder it is to make changes that are not superficial.

Linear life cycles are most appropriate where the extent of scope is relatively simple and predictable, uncertainty of scope is low and the main objective is to be delivered complete at the end of the work. As the extent and uncertainty of scope increase, different approaches are needed.

Incremental

Very often it is advantageous to deliver the outputs incrementally. For example: if the project was to build a housing estate, there would be obvious cash flow advantages to build, and start selling, the first batch before starting the second batch, and so on.

It may be that only the delivery is split into incremental outputs but there may also be benefits to splitting the definition into increments as well, e.g. better cash flow or reduced financial risk if the project is cancelled.

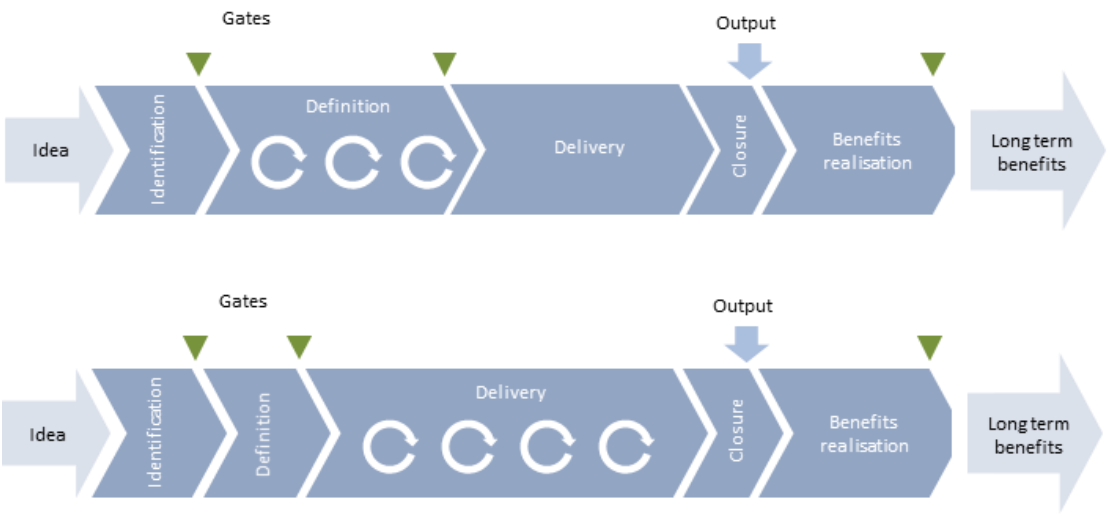


Iterative

Iterative development addresses uncertainty. Instead of trying to develop a full specification before development, iterations progressively develop ideas and prototypes. Developers and customers work together to refine these through multiple iterations until the product is ready for use.

This has always been common in the definition phase as plans and specifications are developed iteratively. Multidisciplinary teams produce high level designs and then work on the detail which may result in issues that are addressed by reworking the high-level designs. Concurrent Engineering was developed in the late 1980s to formalise this approach.

What needs to be produced and how it will be produced are refined iteratively, often through the use of prototypes. This is most appropriate where physical outputs are involved and the cost of change increases rapidly once delivery work starts (e.g. construction and engineering).



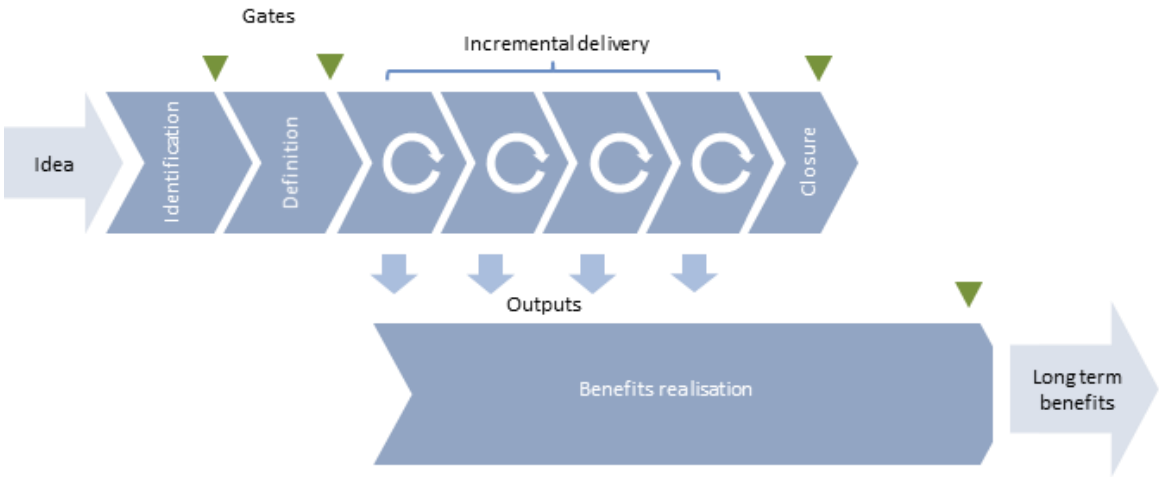
Where the outputs are non-physical (e.g. IT and Business change), the cost of change remains comparatively low through the delivery phase. Since these types of outputs are also more difficult for stakeholders to visualise early on, it makes sense to postpone as much design and specification as possible into the delivery phase.

This greatly increases the opportunity for stakeholder involvement and therefore stakeholder satisfaction. However, it reduces the work done in the definition phase, so overall costs and timescales are less well understood at the point where a decision to proceed with delivery is made. For this reason, organisations have to have a risk appetite that is compatible with this approach.

The simple principles of iterative development were established by the Shewhart Cycle in 1939.

Incremental and iterative

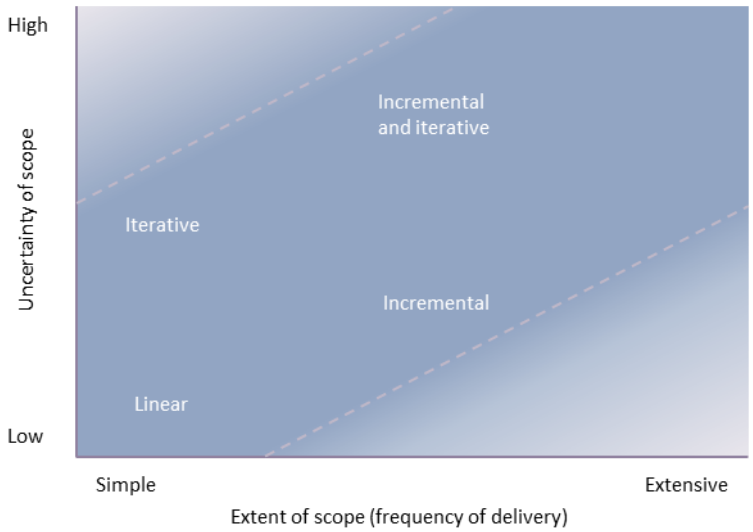
As complexity increases (i.e. as both the extent and uncertainty of scope increases) iterative and incremental approaches are combined. This is often referred to as an agile or adaptive life cycle.



A life cycle that is both incremental and iterative is best placed to deal with complexity. Being iterative is like taking small steps to cross hazardous terrain in fog and frequently reviewing the path. Being incremental, ensures that benefits are gained from sections of the journey and not waiting until the end.

It is important to note that any given project does not have to employ only one of these approaches in its entirety. A large project may involve elements that are best delivered linearly, some that are best delivered incrementally and some that are best developed iteratively. These combinations will be different each time to match the context of the project.

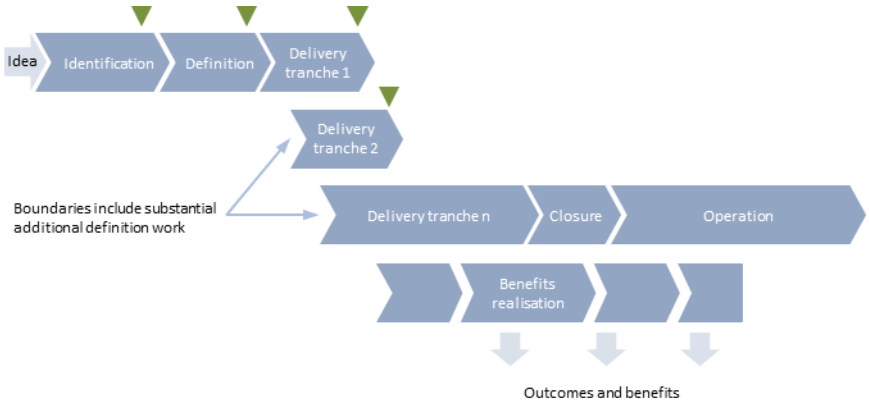
The diagram below shows how the four types of life cycle fit into the Praxis Project Delivery Model.



As the extent of the scope increases, a programme governance approach is more appropriate. This does not require a fundamentally different life cycle but there are a few different conventions.

Programme life cycles are inherently incremental. The delivery phase is broken up into tranches which should each be able to deliver a section of the blueprint and associated benefits.

Programmes are also inherently iterative. The wide scope of a programme makes it virtually impossible to perform a complete design during the definition phase which will only look at the first tranche of projects in detail and outline subsequent tranches. Further detail will be developed at the boundaries preceding each tranche. This definition work will take all previous work into account so that the way to achieve the vision evolves throughout the programme.

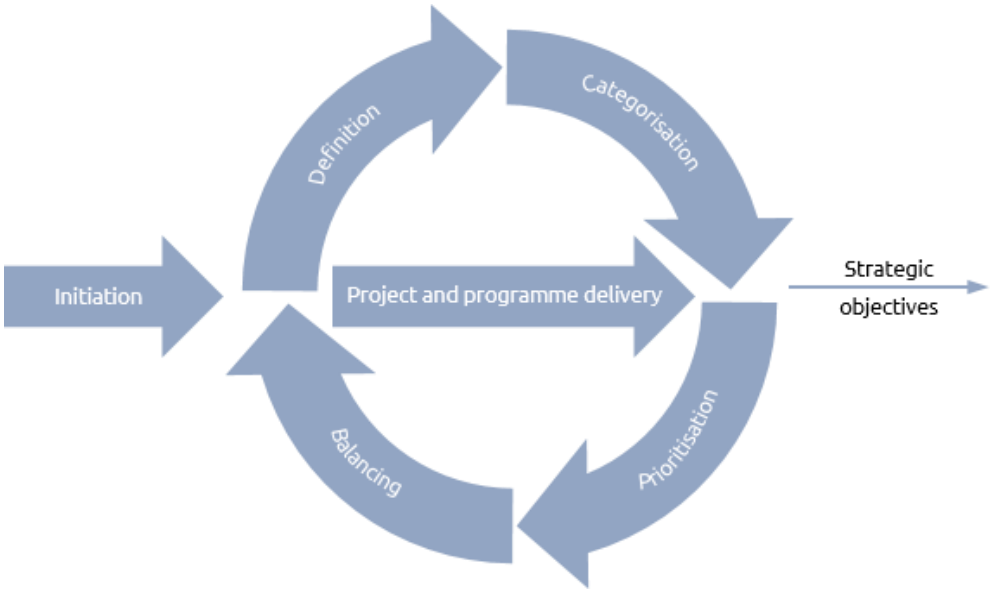


- Identification – the vision and outline business case for the programme are created in this phase. A sponsor is appointed to oversee the phase and provide a mechanism for approvals. The expected benefits are outlined and a programme brief is prepared. Sufficient analysis must be performed to enable the main stakeholders, led by the programme sponsor, to make two decisions:
    - Is the programme likely to be viable?
    - Is it definitely worth investing in the definition phase?
- Definition – the vision is developed into a detailed description of the end state of the programme, often referred to as a blueprint. The management plans, delivery plans and business case are developed so that the sponsor and key stakeholders can make an informed decision whether to proceed with the programme.
- Delivery – this phase is usually broken into groups of projects called tranches that each deliver beneficial change in their own right. A review at the end of each tranche assesses the continuing justification for the programme. During delivery, projects are identified and designed in accordance with the blueprint.
- Benefits realisation – as new outputs are delivered by projects, transformation work has to be done to ensure new ways of working become embedded in business-as-usual. Benefits will be measured and compared to the baseline in the business case. This phase is segmented to reflect the fact that the change management necessary to realise benefits is not constant and will fluctuate in level.
- Closure – closure of the last projects, closure of budgets and demobilising the programme management and delivery teams.

The realisation of benefits will usually continue after the closure of the programme. Some members of the programme team (typically the programme sponsor and business change managers) will continue in their roles to ensure that benefits are realised as required by the business case.

Portfolio life cycle

Unlike projects and programmes, portfolios are less likely to have a defined start and finish. Portfolio management is a more continual cycle co-ordinating projects and programmes. It may, however, be constrained by a strategic planning cycle that reviews strategy over a defined period. If an organisation has, for example, a three-year strategic planning cycle, then the portfolio cycle will have compatible time constraints.



The aim of the portfolio is to co-ordinate projects and programmes and ensure they remain consistent with organisational strategy.

- Initiation – This is a one-off phase that represents the point at which the host organisation decides to set up a portfolio. It is where the infrastructure is created that enables the portfolio cycle to operate.

The portfolio life cycle is inherently incremental with projects and programmes regularly contributing to the required strategic benefits.

At any point in time the emphasis may be on one phase or another, but aspects of all will be undertaken simultaneously.

- Definition – the projects, programmes and change to business-as-usual required to meet portfolio objectives are identified and evaluated in a selection process that maximises the effectiveness and efficiency of the portfolio.
- Categorisation – the projects and programmes may be organised into 'sub-portfolios' or groups that share certain characteristics, such as alignment with particular strategic objectives.
- Prioritisation – priorities can be set by strategic objective, return on investment or any other chosen metric. On the assumption that no organisation has sufficient

In the Praxis process model these four parallel phases are combined within the portfolio management process.

resource to do everything it wants, the prioritisation process forms the basis of the next phase.

- Balancing – the portfolio must be balanced in terms of risk, resource usage, cash flow and impact across the business.

Portfolio management incorporates the overall governance of projects and programmes within the host organisation. The portfolio management team may be responsible not only for co-ordinating the projects and programmes to deliver strategic objectives, but also for improving the maturity of project, programme and portfolio management.

1. Patel, M. B. & Prof. P.G.W. Morris, Guide to the Project Management Body of Knowledge, Centre for Research in the Management of Projects, University of Manchester, 1999.