A Model for Resource Planning HTML5 Gantt Charts

NETRONIC

Table of Contents

Why this Model for Resource Planning?

Over the past few years, we at NETRONIC have supported many of our clients develop web applications for scheduling or resource planning by interactive HTML5 Gantt charts. While we initially developed for each customer tailored widgets based on a special framework we developed ourselves, we are now providing our customers with the powerful [Visual Scheduling Widget](#) (VSW for short). This widget fulfils almost all requirements our customers face in their daily work.

Analogously, for customers living in the world of Microsoft Dynamics 365 Business Central we offer the [Visual Scheduling Add-in Developer Toolbox](#) with nearly the same functionality as VSW provides.

The main scope of application of our HTML5 Gantt charts is some kind of resource planning in different application fields like project management, production scheduling, and supply chain management. Therefore, both products are based on the same model that is specific to resource planning.

This new model is the result of our experiences and intensive cooperation with our customers. It includes not only a new data model for resource planning, but it also provides a concept of how to map the members of this data model to an interactive graphical representation.

To quickly become familiar with the concept of our products and their programming interface (API), we recommend studying the following brief introduction to the model.

In this document you will find some UML diagrams that illustrate at a glance the relationships between the object type currently under consideration and the associated ones**. Only those object properties are listed that are essential for understanding the concept of this data model**. The **dark blue shaped types** in these diagrams are the ones that can be processed with the methods add..., update... and remove... of the widget. To do this, they explicitly provide identifiers in the form of the ID property. In contrast, the **light blue types** include dependent objects without their own identifiers.

Teammates of Resource Planning Data Model

The following list provides a detailed overview of the different types of objects involved in the Resource Planning Data Model and their relationships to each other.

Activities and Activity Entries
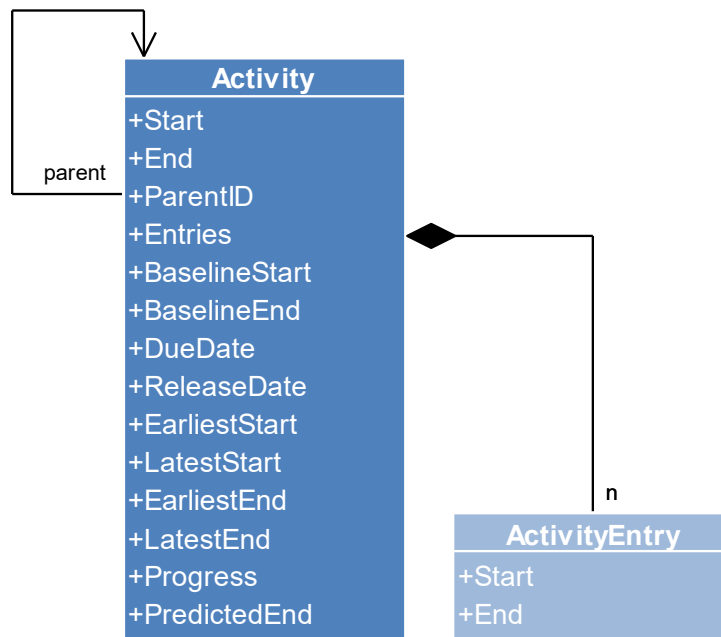
The term Activities covers all actions that need to be accomplished within a certain time period to achieve a specific objective. Examples of activities are tasks in project management or jobs and operations in production planning.

Start and end dates of an activity are the two important properties that can be the result of a scheduling process or that are predefined.

To build hierarchical structures, activities can be nested. For instance, in project management a project might be divided into many tasks and each task consist of a set of subtasks and so on. In production planning, jobs are usually subdivided into further jobs and on the lowest level operations are the base activities. Nesting can be achieved by setting the identifier of a parent activity.

Properties such as progress, special types of constraint dates, and baseline dates are also considered in the data model.

Usually, an activity is represented by a homogeneous continuous bar beginning at the start date and ending at the end date. But imagine the execution time of an activity is subdivided in different time

spans. For instance, if the execution time consists of a setup time, a pure runtime, and a teardown time, then it may be desirable to visually separate these time spans. For this purpose, an activity can have an array of activity entries. Each entry has its own start and end date and represents a time span that can be customized by graphical properties. In the above case, we could define an entry each span.
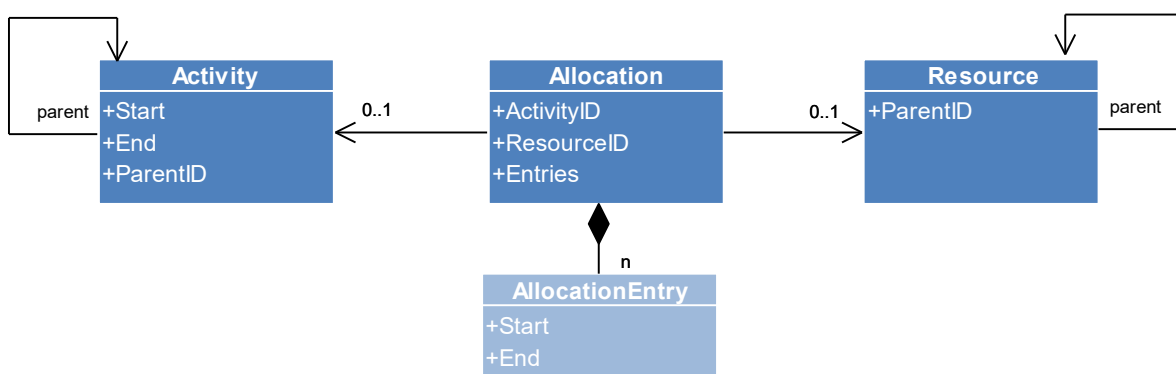
Resources
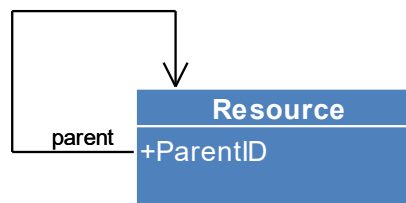
Resources – such as machines or personnel – are the ones that are needed to perform the activities. Besides its general availability defined by a calendar, a resource has a capacity and a load. The latter results from the assignment of activities to a resource and is defined by specific curve objects.

As well as with activities, resources might be nested. Departments consisting of groups of individual resources are examples of a three-level nesting (department-> resource group->resource).

Allocations and Allocation Entries

Allocations define the assignments of resources to activities and vice versa. Each allocation refers to a maximum of one activity and a maximum of one resource

In the simplest case, a resource is allocated by an activity for the entire execution period of the activity. Then the allocation will use the start and end dates of the activity itself. But imagine the

execution time of an activity is subdivided in different time spans. For instance, if the execution time consists of a setup time, a pure runtime, and a teardown time, then a resource may only be required for setup and teardown, but not at runtime. For this purpose, an allocation can have an array of allocation entries. Each entry has its own start and end date and represents a time span. In the above case, we will define an entry for the setup time and another one for the teardown time.

Of course, this concept of allocations also allows to model many-to-many relationships between activities and resources by using multiple allocations with the same activities and/or resources involved.

Calendars and Calendar Entries
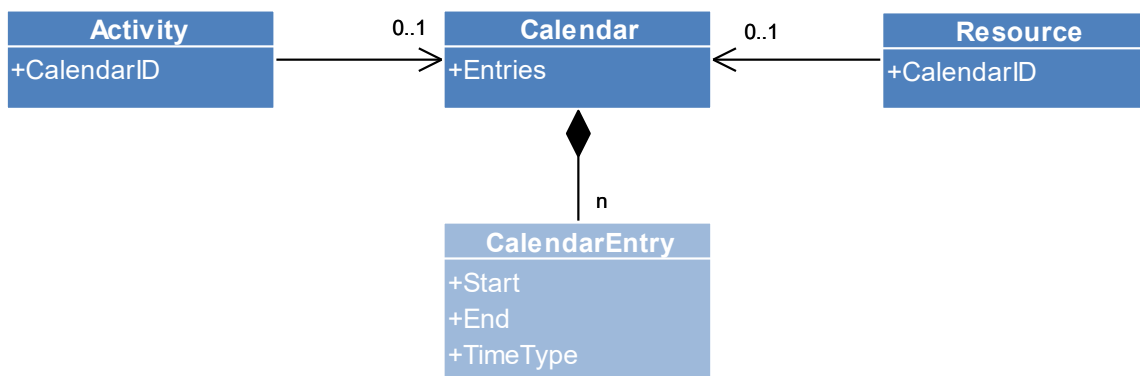
As mentioned above, calendars are used to define the general availability of a resource without considering the specific capacities. Each calendar has an array of calendar entries that determine whether it is a working or a non-working time for a time period specified by a start and an end.

Calendars can be referenced by activities as well as by resources.

Curves and Curve Point Entries
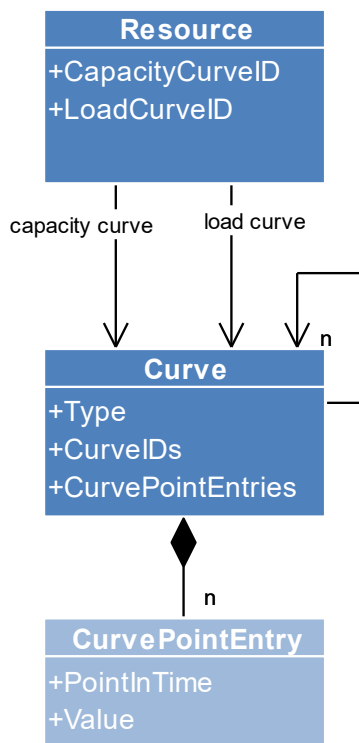
Curve objects are used to handle time-oriented data such as capacities or loads of resources. They can be simple point curves with an array of curve point entries, each entry defining a point in time and a corresponding value. Or they can be stacked curves that are built by adding multiple curves. In this way, you can, for example, add all capacities of individual resources in a resource group to a total capacity of this resource group.

To make working with curves easier, you only need to define entries for those points at which the curve changes its value.

Links
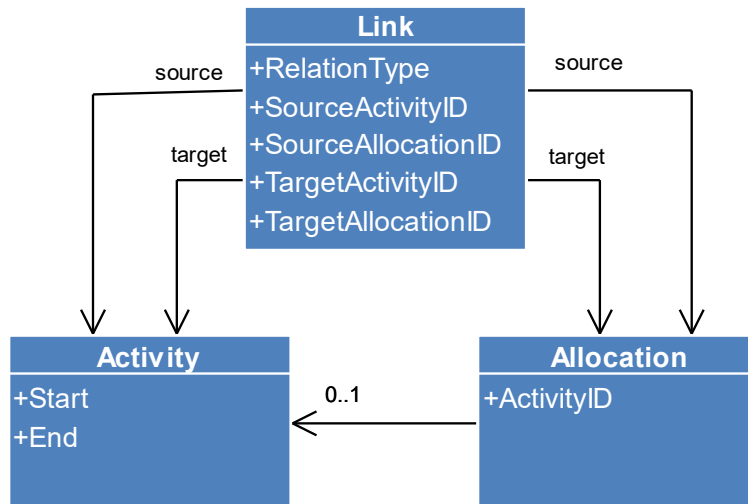
In addition to grouping, activities can be related to each other in a different way. Think, for example, of chained production operations or of complex networks project management tasks. In this case, an activity can only be started if the directly preceding activity has already been finished. Such a kind of relations can be expressed by link objects.

Each link is directed and refers to a source and a target activity. In addition to these references, a link has one of the well-known relation types (finish-start, start-start, finish-finish).

Since you can create multiple links with the same source or the same target activity, not only simple activity chains, but also more complex networks of activities can be realized.

Entities
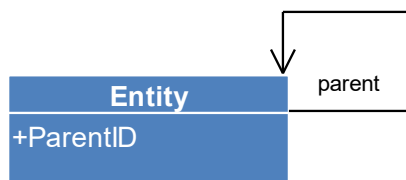
This type of objects is used to manage arbitrary sets of any grouped objects. An entity can be an activity, a resource, or any other object. In practice, entities are used, for instance, for lists of backlog orders that have not yet been scheduled or for lists of available resources with suitable skills that can be assigned to a certain activity.
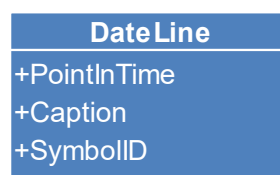
As with activities or resources, entities can be nested to build hierarchical structures.

Table Row Definitions and Table Cell Definitions

```
┌──────────────────────┐   ┌──────────────────────┐   ┌──────────────────────┐
│       Activity       │   │       Resource       │   │        Entity        │
├──────────────────────┤   ├──────────────────────┤   ├──────────────────────┤
│ +PM_TableRowDefinitionID │ +PM_TableRowDefinitionID │ +PM_TableRowDefinitionID │
└──────────────────────┘   └──────────────────────┘   └──────────────────────┘

                        ┌──────────────────────┐
                        │  TableRowDefinition  │
                        ├──────────────────────┤
                        │    +CellDefinitions  │
                        └──────────────────────┘
                                   ◆
                                   n
                        ┌──────────────────────┐
                        │  TableCellDefinition │
                        ├──────────────────────┤
                        │    +TextSource       │
                        │    +Title            │
                        └──────────────────────┘
```
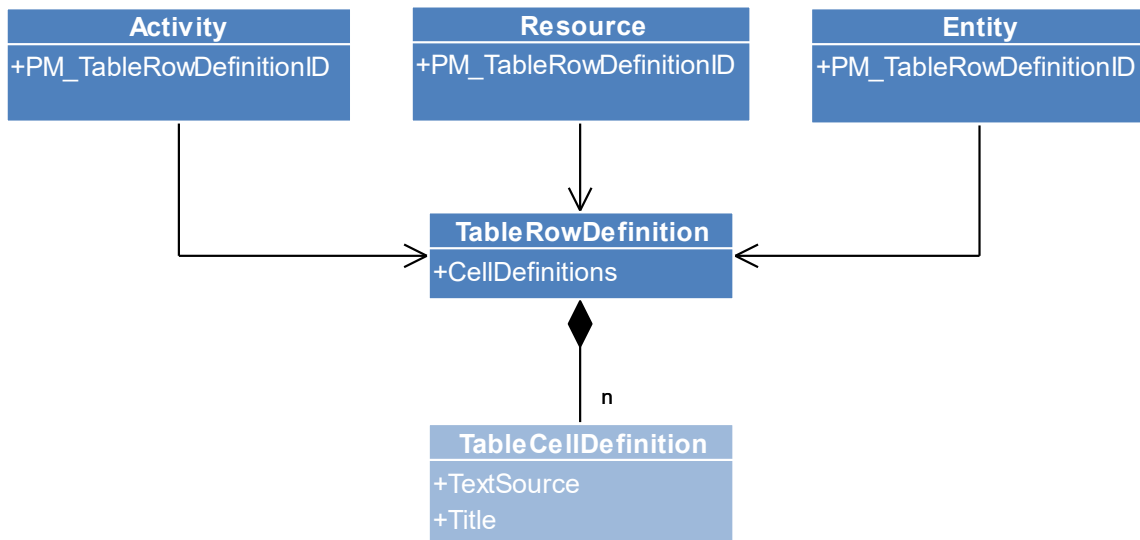
To list activities, resources, and entities in tables, you need to define the appearance of the tables. This is the task of the table row and cell definitions. A row definition can be assigned to each activity, resource. And each row definition consists of several cell definitions, each of which, for instance, defines the source of the text to be displayed in the table or the width of the cell. Therefore, row and cell definitions are in the stricter sense not part of the pure business data but are used for visualization.

Date Lines

```
                        ┌──────────────────────┐
                        │       DateLine       │
                        ├──────────────────────┤
                        │    +PointInTime      │
                        │    +Caption          │
                        │    +SymbolID         │
                        └──────────────────────┘
```

Date lines are used for visualizing any date/time data in the time area of a planning chart. To leave a semantics, the lines can be labelled.

Symbols
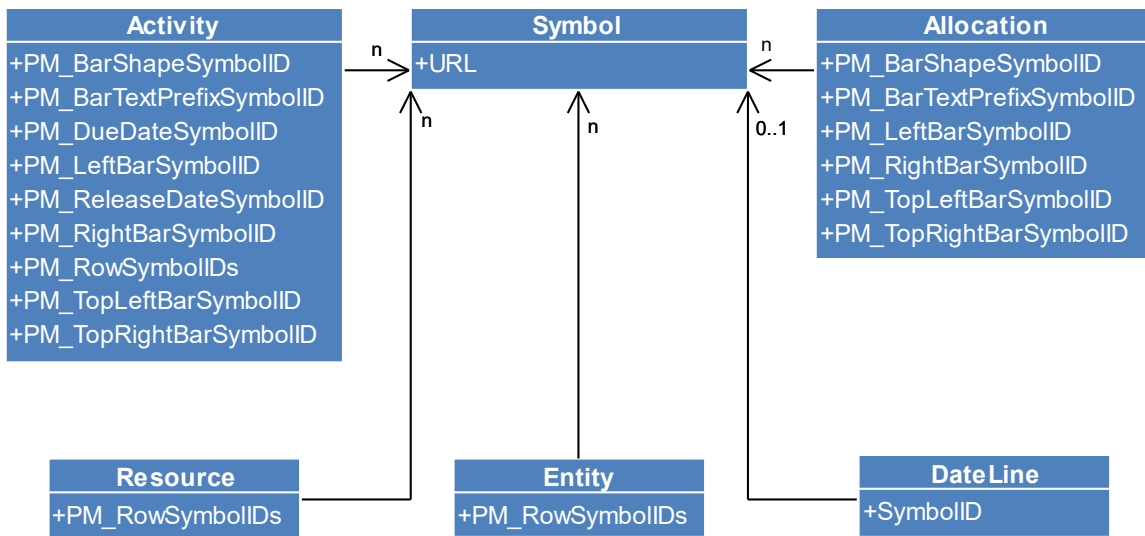
The freely definable Symbol objects are applicable to activities, allocations, datelines, entities, and resources in a variety of ways. They can be used to mark table rows, to be displayed in table cells, or to put additional information to the bars at several predefined positions.

Tooltip Templates
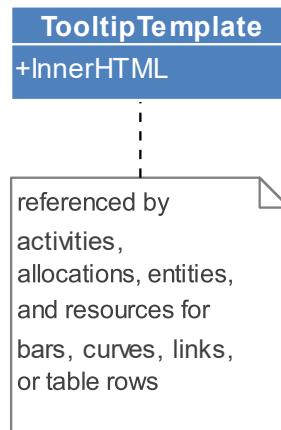
Tooltip templates define the structure and contents of tooltips using an HTML string. They can be assigned to activities, allocations, entities, and resources to supply their graphical representations like bars, curves, linking lines, or table rows with tooltips.

Period Highlighters
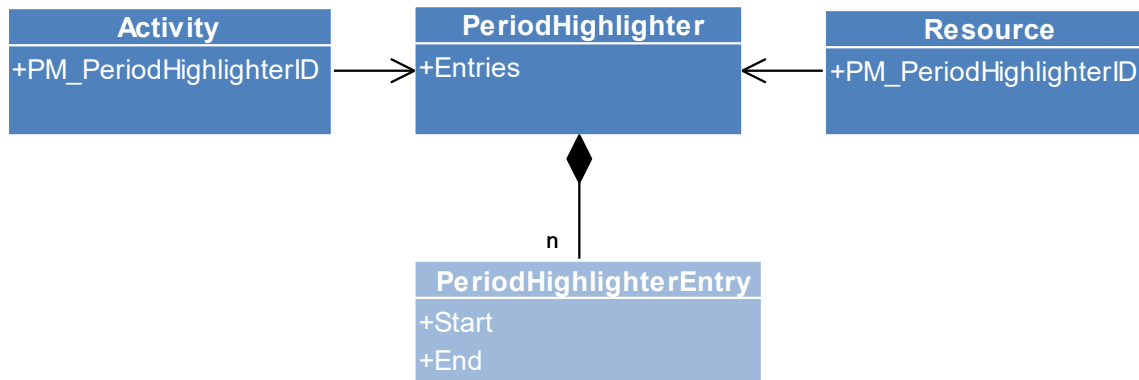
A PeriodHighlighter object is a pure presentation object and defines the properties of a series of time periods that can be shown on each resource row and activity. Each time period can be colored independently and can have a caption of its own. In contrast to the grids created by Calendar objects, the time periods do not define work or non-work times, but only highlight time periods graphically.

Hierarchy Supplementary Definitions
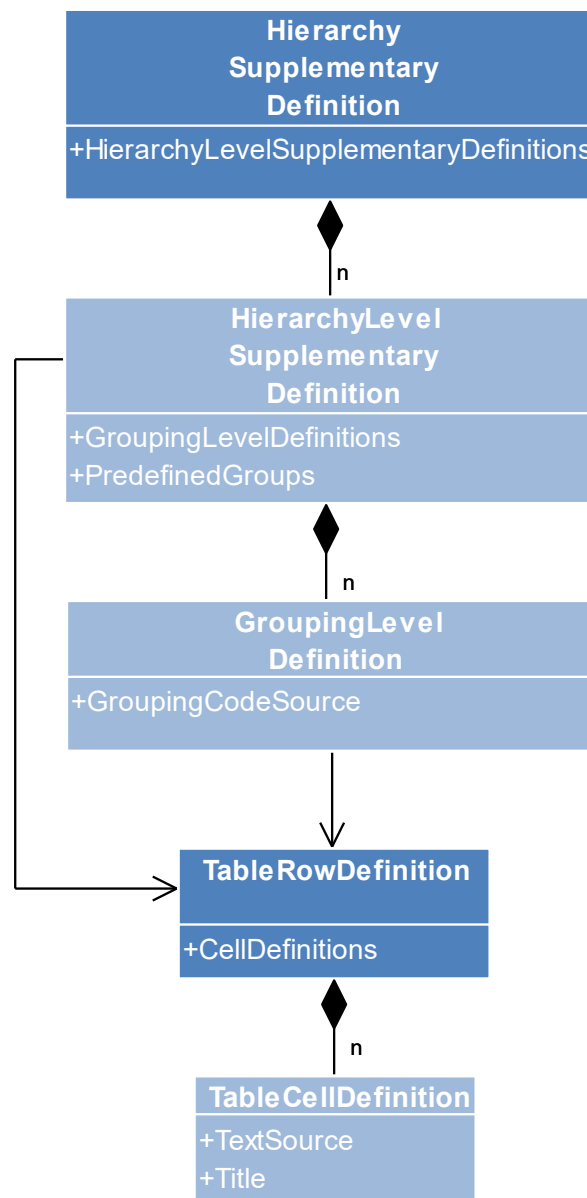
A **HierarchySupplementaryDefinition** object defines for a whole hierarchy – e.g., an activity hierarchy in the Activities View, a resource hierarchy in the Resources View, or an entity hierarchy in the Entities Table – additional information about how the hierarchy items should be grouped.

A hierarchy supplementary definition contains an array of objects – the so-called **HierarchyLevelSupplementaryDefinition** objects – each of them providing additional grouping information for each single hierarchy level, where the first object in this array is responsible for the first level, the second one for the second level and so on.

The grouping information for a hierarchy level contains an array of **GroupingLevelDefiniton** objects, each specifying the source from which the grouping criteria should be taken. This way, each hierarchy level can be structured by a multi-level grouping.

It is possible to define multiple **HierarchySupplementaryDefinition** objects so that different hierarchies can be specified and applied to different views, or the hierarchy structure of a view can be altered by selecting the appropriate definition.

Map to Interactive Graphical Representations

Having a data model is nice, but what is needed at the end is an interactive chart that allows a user to work with that data in the most intuitive way possible. Therefore, an adequate representation for all members of the data model in form of a Gantt diagram has to be found.

In practice, two types of charts have proven to be very useful and valuable:

- the activities view and
- the resources view.

One can imagine them as the two sides of the same coin. Both present the data model from a different perspective with different focus.

In order to be able to specify all desired aspects of the mapping of the data model to interactive, graphical representations, many additional properties have been added to each object type of the data model. These properties – they are marked by the prefix "pm_", which stands for "presentation model" – can be used to control the following aspects of the presentation:

- colors of graphical elements such as bars, table rows, lines, texts, timescale, calendar grids, …
- collapse status of rows/groups (expanded, collapsed)
- possible interactions (whether a bar can be moved or resized, whether a row can be collapsed or not, which elements are selectable, snapping targets, etc.)
- design of expanded and of collapsed rows (whether and how child elements should be displayed)
- visualization of special properties such as constraint dates, status, symbols, etc.

Many of these properties can be set both as default values and as individual values for individual objects. For further details, please refer to the Interface Definition Document – IDD for short – of the respective widget.

The Activities View

In this view, activities are listed in the table area of the Gantt diagram. In the simplest case, each activity has its own row. In the diagram area, the start and end dates of the activities are represented by bars. Resources and allocations are not of interest in this view type.

If activities are grouped, then the parent activities can be expanded:
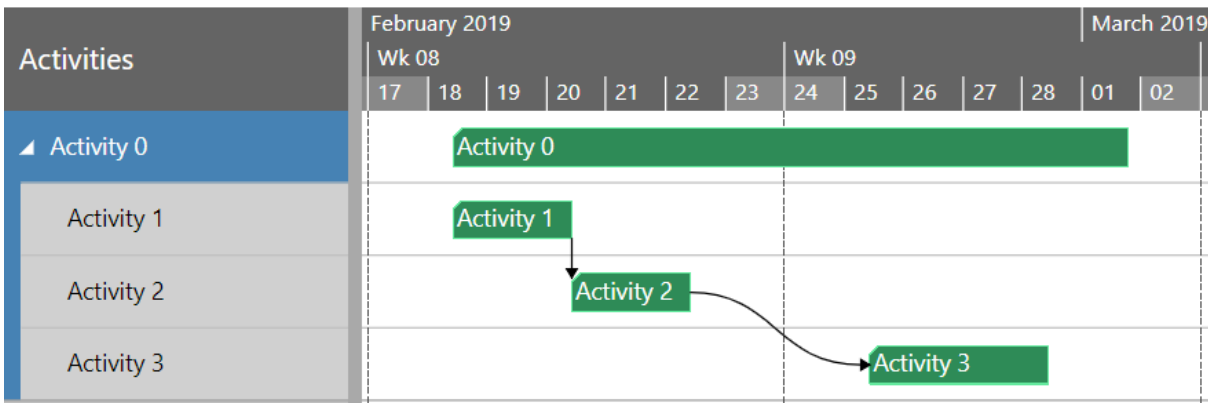
Or the parent activity can be interactively collapsed:
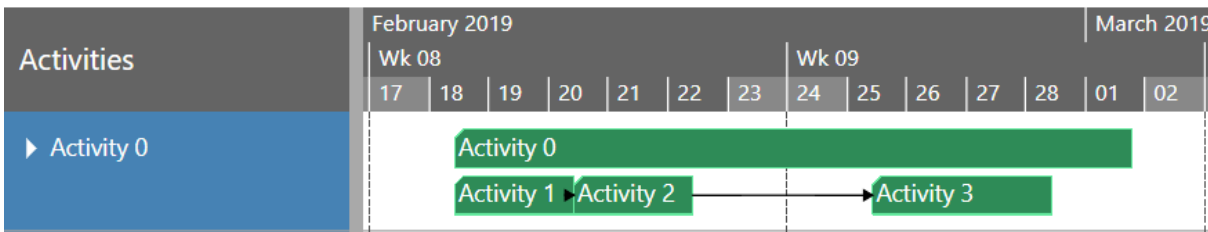
In the collapsed state, the parent activity itself and all child activities are positioned in one row. But this is only one possible way we provide to display collapsed rows. Alternatively, you can choose a different design and decide to show the parent activity only:
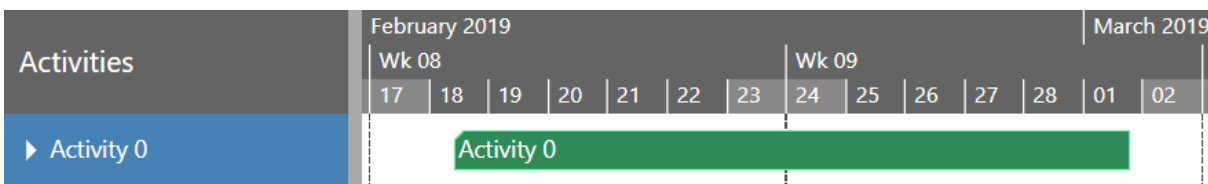
Or instead of the parent only the children are to be seen:
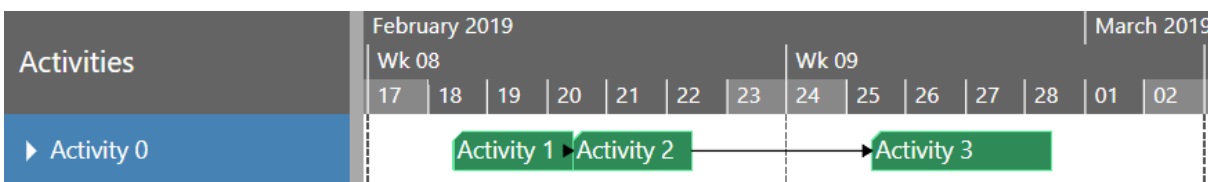
In any case, the concrete design for collapsed activities can be defined individually for each activity as well as globally by default settings.

Link objects are mapped to arrows between activity bars. The relation type of a link is expressed by the layout of the arrow, which can begin and end at the beginning or the end of a bar depending on the type.

The Resources View

Analogous to the activities view, the resources view lists all resources in the table area of the Gantt diagram. In the diagram area the allocations and their entries are displayed. The calendar entries for non-working times of a resource are visualized by slightly grayed vertical strips.
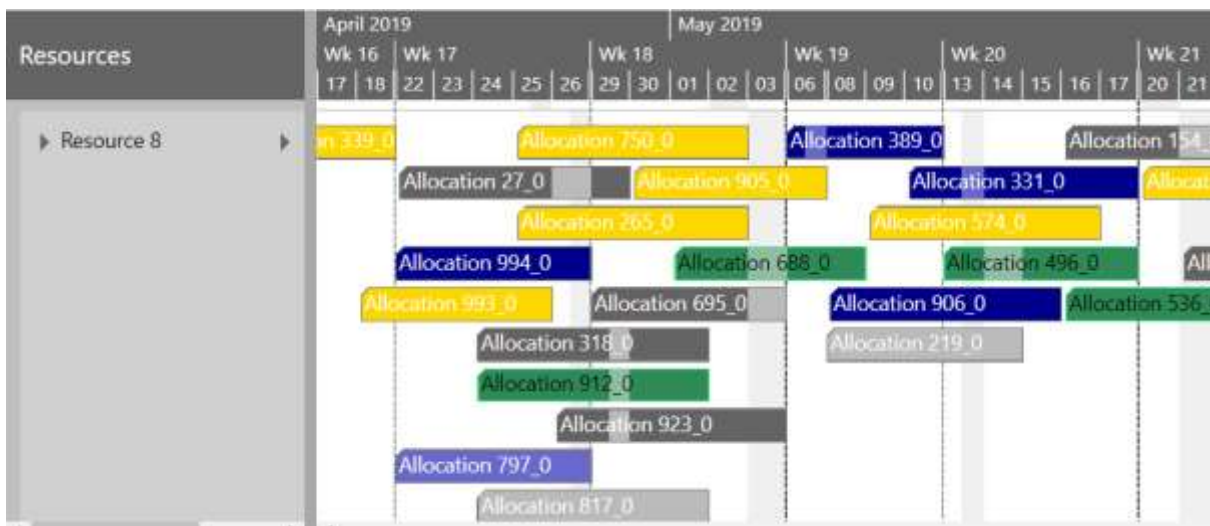
There are multiple designs for visualizing allocations.

- An allocation can be represented by a single bar:

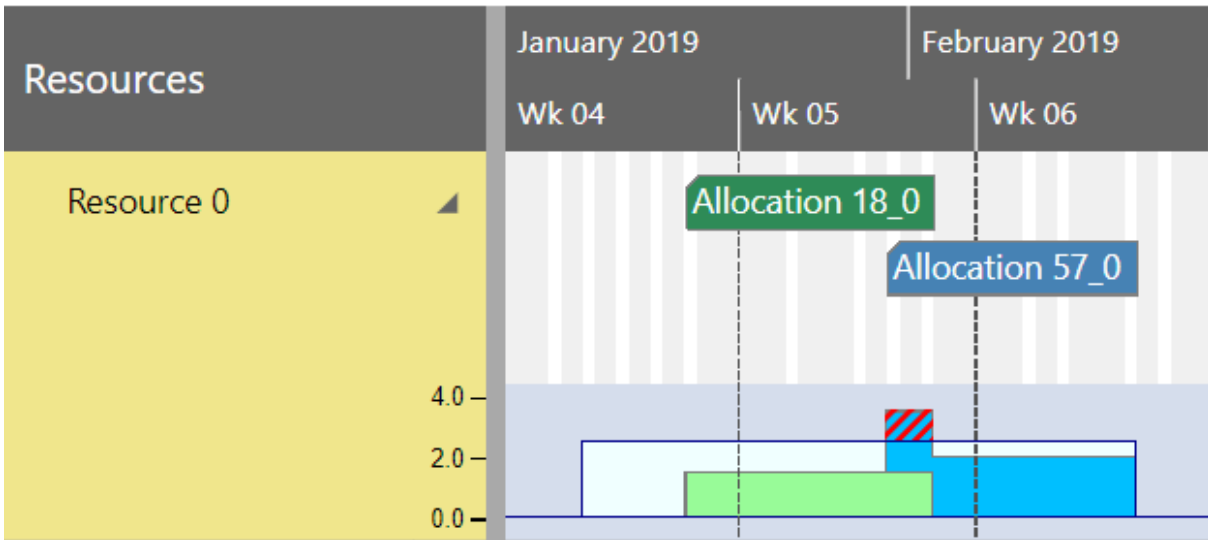- If it has more than only one entry, then each entry can be displayed as a separate bar:

- Or the entries are shown together with the allocation itself:

For displaying curves there is inside each resource row a curve area, which can be interactively expanded and collapsed.

A curve object is displayed as a stepped line, possibly with the colored area below the line. Each curve entry defines a data point of that stepped line.

With stacked curves, not only the stack as a whole is visible, but also the single members of the curve stack.

The following figure shows two curve objects, each of them representing the load of a resource caused by an allocation. They are stacked on top of each other and visually distinguishable by two colors. An additional curve object defines the capacity of the resource and is displayed by a line. The area where the stacked curve exceeds the capacity line is highlighted by hatching:
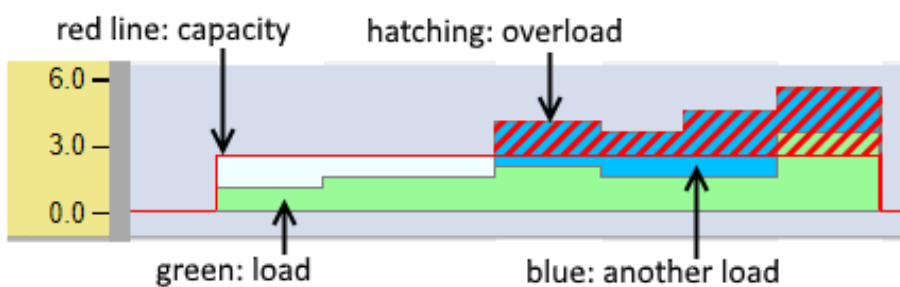
Of course, this is only one possible sample for the use of curves. A lot of further applications for curves are conceivable and you can use curves to express any other semantics.

The Loads View

Last but not least, there is an additional view type – the loads view – that is a derivate of the resources view with the curve diagram areas only and without any bars. It is a valuable tool when only the capacities and loads of the resources are of interest, but not the allocations in detail.

The Entities Table

The entities objects are displayed in a special table – the entities table – that can be activated to the right of the Gantt diagram.