



## Cas d'usage

On récupère un tableau d'affectation de techniciens pour des interventions (installation, livraison, réparation) chez des clients. On désire obtenir une représentation graphique du planning des techniciens (considérés ici comme des ressources). Nous allons utiliser un *chart* de type Diagramme de Gantt proposé dans Oracle APEX et nous allons voir comment construire la requête .

Abstract: Oracle APEX - Gantt Chart with multiple tasks per row

## Échantillon du résultat qui sera obtenu

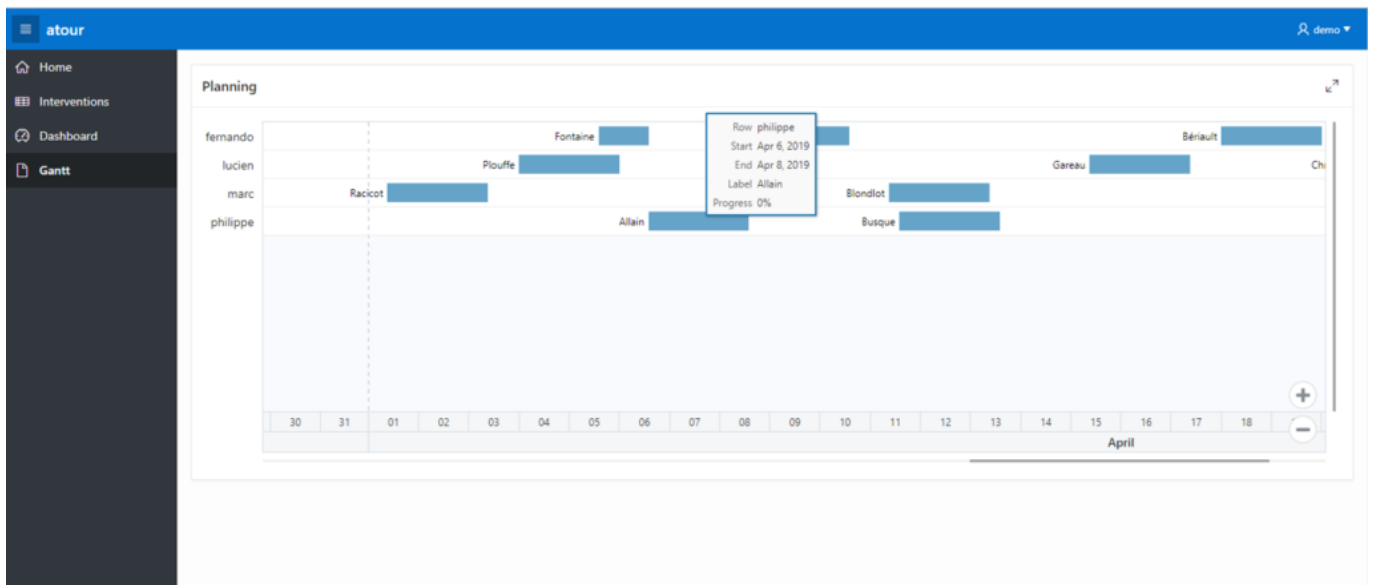


Diagramme Gantt de ressources



## Structure de la table

la colonne NOM contient le nom du client.



TECHNICIEN contient le nom du technicien.

Column Name	Data Type
ID	NUMBER
CIVILITE	VARCHAR2(30)
PRENOM	VARCHAR2(30)
NOM	VARCHAR2(30)
ADRESSE	VARCHAR2(30)
VILLE	VARCHAR2(30)
CODE_POSTAL	VARCHAR2(30)
EMAIL	VARCHAR2(255)
TELEPHONE	VARCHAR2(30)
LATITUDE	NUMBER
LONGITUDE	NUMBER
DEBUT	DATE
TECHNICIEN	VARCHAR2(30)
DUREE	NUMBER
GOLD	VARCHAR2(1)

Échantillon du contenu de la table



	Civilite ↑	Prenom	Nom	Adresse	Ville	Code Postal	Email	Telephone	Latitude	Longitude	Debut	Technicien	Duree
	M	Ferragus	Bériault	90 Rue Joseph Vernet	BAGNEUX	92220	FerragusBeriault@gustr.com	01.37.89.60.98	49	2	18-APR-19	fernando	2
	M	Lance	Racicot	74 Place de la Madeleine	PARIS	75010	LanceRacicot@jourrapide.com	01.59.14.22.55	49	2	01-APR-19	marc	2
	M	Théophile	Bonenfant	82 rue de l'Aigle	LA MADELEINE	59110	TheophileBonenfant@fleckens.hu	03.38.01.93.96	51	3	21-APR-19	-	2
	M	Malagigi	Berthiaume	68 rue de la Boétie	PLAISIR	78370	MalagigiBerthiaume@fleckens.hu	01.30.51.75.86	49	2	03-MAR-19	lucien	3
	M	Franck	Jomphe	27 Avenue des Pries	MONTLUÇON	3100	FranckJomphe@einrot.com	04.30.28.09.46	46	3	13-MAR-19	-	2
	M	Zerbino	Cantin	39 rue de Geneve	ALFORTVILLE	94140	ZerbinoCantin@teleworm.us	01.04.98.29.28	49	2	21-APR-19	fernando	2
	M	Thierry	Deniger	3 rue Adolphe Wurtz	LE ROBERT	97231	ThierryDeniger@dayrep.com	05.17.66.77.83	46	5	-	-	-
	M	Henry	Busque	19 Square de la Couronne	PARIS	75002	HenryBusque@fleckens.hu	01.70.86.30.99	49	2	11-APR-19	philippe	2

Échantillon du contenu de la table

## Query dans la section « Series »

L'objectif est de construire un tableau dont les lignes correspondent aux techniciens et les taches correspondent aux clients. On fabrique un id artificiel pour chaque technicien en calculant une valeur de hash et on combine (opérateur UNION) la liste des intervention avec celle des clients de façon à produire une arborescence de taches à un niveau. La fonction `ORA_HASH` appliquée au nom du technicien produit un ID utilisé comme `Parent_id` des interventions.

```
select * from
(
select distinct technicien task_name,
ora_hash(technicien)          task_id,
```



```

null parent_task,
null task_start_date,
null task_end_date,
0 status,
null gold,
(select min(debut) from atour) gantt_start_date,
(select max(debut + duree) from atour) gantt_end_date,
null viewport_start,
null view_port_end
from atour
where technicien is not null
UNION
select gold||'_' || nom task_name,
id task_id,
ora_hash(technicien) parent_task,
debut task_start_date,
(debut+duree) task_end_date,
0 status,
gold,
(select min(debut) from atour) gantt_start_date,
(select max(debut + duree) from atour) gantt_end_date,
(sysdate -10) viewport_start,
(sysdate + 10) view_port_end
from atour
where debut is not null
and technicien is not null
)
start with parent_task is null
connect by prior task_id = parent_task
order siblings by task_name
```



## Code Javascript dans la section « Attributes » pour l'affichage des taches

Optionnellement, on peut aménager l'affichage pour produire des effets. Ici, on demande l'affichage d'une ligne rouge qui matérialise la journée en cours et on applique une couleur spécifique de tache pour les clients possédant un contrat de type « gold ». En raison de limitations des attributs passés dans le payload json, on concatène la valeur du champs GOLD avec celle du champs NOM (cf requete précédente) pour ensuite demander un filtrage u niveai Javascript. cf [documentation Oracle Jet pour le composant Gantt](#).

```
function( options ){
    options.dataFilter = function( data ) {
var Mydata = data.rows;
    Mydata.forEach(function(nrow, i) {
        nrow["tasks"].forEach(function(nr, k) {
            if (String(nr.label).substring(0,1) == '1') {
                // data.rows[i].tasks[k].svgStyle      = {'fill':
'blue'};
                // data.rows[i].tasks[k].svgStyle      = {'line':
'blue'};
                data.rows[i].tasks[k].svgClassName = "u-color-20";
                // data.rows[i].tasks[k].stack-label-style =
{'fill': 'red'};
            }
            data.rows[i].tasks[k].label =
String(nr.label).substring(2);
        });
    });
    return data;
}
```



```
};  
  
var event = new Date();  
// Define Reference Object line on the chart  
var constantLine = [ { value: event.toISOString() } ];  
  
// Set the referenceObjects option as part of the chart initialization  
options.referenceObjects = constantLine;  
  
//options.gridlines.vertical = "visible";  
return options;  
  
}
```