# Title: Application Performance Enhancements

**Current Request Pipeline**

### Leads/Contacts/Accounts pages

Currently, the application request pipeline is as follows:

1. Application bootstrap
   1. Session
   2. Integration
   3. Forms
   4. Lead Stages
   5. Deal Stages
2. Route
   1. Filters/Views/Segments
   2. Saved Filters
3. Controller
   1. Filter Options
   2. Entity Info (Leads/Contacts/Accounts)

The requests in each layer (application bootstrap/route/controller) are in most cases parallel. However, the pipeline between layers is sequential. This essentially means that the request to fetch the most useful data to the end-user is the last to get triggered. This results in a pretty significant delay.

**Proposed changes to the request pipeline**

### Leads/Contacts/Accounts pages

##### Phase 1

###### First Time application load

1. Application bootstrap
   1. Session
   2. Integration
   3. Forms
   4. Lead Stages
   5. Deal Stages
2. Route
   1. Filters/Views/Segments *(Required to fetch default segment identifier)*
3. Controller
   1. Entity Info (Leads/Contacts/Accounts)
   2. Saved Filters (Background load)

###### Subsequent application load

1. Application bootstrap
   1. Session
   2. Integration
   3. Forms
   4. Lead Stages
   5. Deal Stages
2. Controller
   1. Entity Info (Leads/Contacts/Accounts) *(segment identifier loaded from local cache)*
   2. Filters/Views/Segments
   3. Saved Filters (Background load)

The requests in each layer (application bootstrap/route/controller) are in most cases parallel, except the request for "Saved Filters". The request for "Saved Filters" is triggered as a background task after entity and views requests are processed. The necessary information for fetching  
  
The request for "Filter Options" is triggered only when the user activates the "Filters" panel (Right panel)

##### Phase 2

###### First Time application load

1. Application bootstrap
   1. Session
   2. Integration
   3. Forms
   4. Lead Stages
   5. Deal Stages
2. Route
   1. Filters/Views/Segments *(Required to fetch default segment identifier)*
3. Controller
   1. Entity Info (Leads/Contacts/Accounts)
   2. Saved Filters (Background load)

###### Subsequent application load

1. Application bootstrap
   1. Session
   2. Integration
   3. Forms
   4. Lead Stages
   5. Deal Stages
2. Controller
   1. Entity Info (Leads/Contacts/Accounts) *(segment identifier loaded from local cache)*
   2. Current Segment Information
   3. Saved Filters (Background load)

The request for "Filters/Views/Segments" on a subsequent load is triggered only when the user activates the "Views" panel (Left panel)

**Performance Metrics**

Post  Unable to render Jira issues macro, execution error.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Account | Average Performance Gain (in Seconds) | | | Highest Recorded Performance Gain (in Seconds) | | |
| Leads | Contacts | Accounts | Leads | Contacts | Accounts |
| Freshdesk | 1.8 | 0.6 | 0.5 | 3.6 | 0.8 | 0.8 |
| Mimeo | 0.8 | 0.4 | 0.4 | 2.1 | 1.2 | 1.5 |
| RenMoney | 0.6 | 0.4 | 0.4 | 1.9 | 0.5 | 0.4 |
| ChargeBee | 0.6 | 0.4 | 0.3 | 1.8 | 0.5 | 0.4 |

The metrics stated above reflects the average time (over 20 attempts) to render actionable data on the viewport.