

RMIT University

Assignment 2

iPhone Application “Trakx”



Name: Matthew Croxson

Student Number: S3240944

Unit Name: iPhone Software Engineering

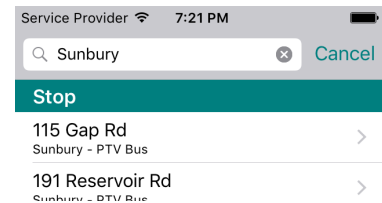
Date Submitted: 21/08/2016

Design	3
Core Data Implementation	4
REST Implementation	5
Transport Disruptions	5
Stops Nearby	6
Stopping Pattern	6
Search	7
Stops on a Line	7
Next Departures	8
Cocoa Framework Implementation	9
Advanced Feature	10
Size Classes	11

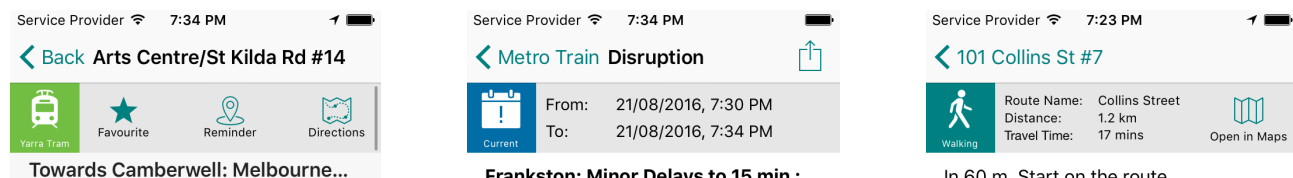
Design

Trakx has been developed with C.R.A.P design principles in mind and as a result provides a coherent user experience across all views. From the outset Trakx has had a singular highlight colour (R: 0, G:127, B:127) used in the application icon and throughout the app. This colour is used on a light background throughout the app to provide a contrast against the background itself, and also non-interactive components of the views.

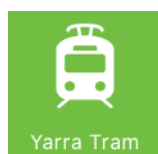
There are a couple of areas of Trakx where the colour has been used as a generic highlighting where the highlight is not related to a transport mode (and hence the colour associated with it). For example, in the table view section headings on the search table view controller the Trakx green is used as a background colour to help contrast the headings against the results themselves (see screenshot right). When used in this manner the green is always the background colour, with white text to allow for readability and to ensure the user doesn't misinterpret the item as an interactive component.



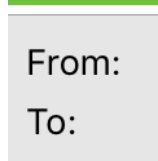
Throughout the app wherever a table view header has been used the styling is identical, although the items within may differ. This includes height and general layout, as shown below:



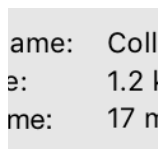
All text displayed within the application is aligned with similar grouped items. Using the same screenshots above there are three examples of this alignment:



Status Icon: Text in the status icon is dynamic across each view with varying length. This text is centred within the label, which itself is centre-aligned with the image above it. This same alignment is shown with the buttons also.



Data Label (Horizontal Alignment): Text in these labels is left aligned, and the labels are left-aligned to each other.



Data Label (Vertical Alignment): Text in these labels is bottom aligned, and the labels are aligned along their bottom edges to.

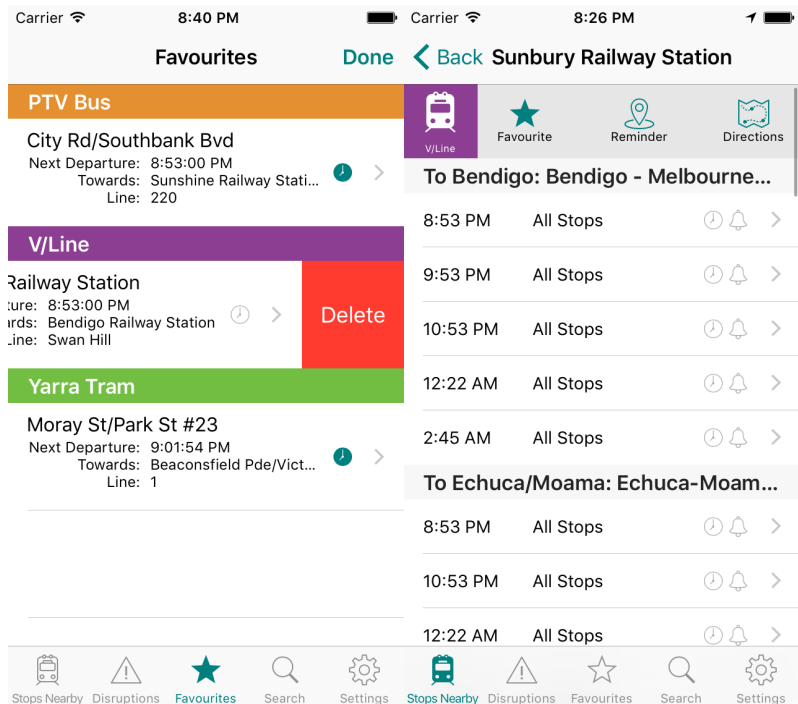
All user interface elements are also designed so as to be separate enough from other elements to be identifiable, but close enough where necessary to imply their relationship. Again using the screenshots above as an example, the data and heading labels are separated so that they don't get confused as a single label, but are also close enough together to ensure the user understands what the data means. There is only one exception to this rule: the stops nearby map. Due to the nature of the response from the PTV API, stop annotations will overlap where there are multiple stops in a single location. Tapping these repeatedly will reveal the annotations underneath.

Route Name:	Collins Street
Distance:	1.2 km
Travel Time:	17 mins

Core Data Implementation

Trakx uses CoreData to allow the user to save particular stops returned by the API as favourites. These favourites are displayed in a tableview on the “Favourites” tab as per the screenshot on the right. The status of a favourite is also visible on the “Stop Detail” view, also shown above.

The data model is relatively simple. There are two entities: FavouriteStop and FavouriteLineType. These entities are connected using a reciprocal relationship, where one FavouriteStop can related to any single FavouriteLineType (one-to-one), while a FavouriteLineType can relate to many FavouriteStop entities (one-to-many). The reason for these relationships is to allow the favourites data table to be displayed in sections by line type, as shown in the screenshot above.



Favourites are added to the data store from the StopDetail view by tapping the Favourite button shown above. Favourites can also be removed by tapping the Favourite button again. The status of the favourite is displayed by either a filled (favourite) or unfilled (not favourite) star icon.

Favourites can also be removed by swiping across the cell in the favourites table view as shown above.

Both views use a shared data model that accesses a NSFetchedResultsController to retrieve, create, update and delete data from the data store. Once changes have been saved relevant delegate functions from the fetched results controller trigger NSNotifications that are received by the interested classes to update their views as necessary.

REST Implementation

Trakx uses data retrieved from the Public Transport Victoria (PTV) Timetable API (details available at <https://www.data.vic.gov.au/data/dataset/ptv-timetable-api>). Trakx retrieves transport disruptions, nearby stops, timetable informations, stops on a transport line, and stopping pattern details, as well as performs searches for stops and transport lines. Sample URL's and JSON responses for each are listed below.

The third-party frameworks Alamofire and SwiftyJSON are used to perform requests to the API and process the JSON responses. SwiftyJSON is also used in the API object classes I have created to greatly simplify code.

Transport Disruptions

Sample URL

<http://timetableapi.ptv.vic.gov.au/v2/disruptions/modes/general,metro-bus,metro-train,metro-tram,regional-bus,regional-coach,regional-train?devid=1000411&signature=16809E314BF6ED21C594047FD9BD0B4B6C1E0ACB>

Sample Response

```
{
  "general": [],
  "metro-bus": [
    {
      "disruption_id": 69453,
      "title": "Minor Delays to 10 min : Route: 234 Garden City - City (Queen Victoria Market) : both directions",
      "url": "http://ptv.vic.gov.au/live-travel-updates/",
      "description": "Minor Delays to 10 min : Route: 234 Garden City - City (Queen Victoria Market) : both
directions",
      "status": "Current",
      "type": "Minor Delays",
      "publishedOn": "2016-08-17T00:32:11Z",
      "lastUpdated": "2016-08-18T22:13:39Z",
      "fromDate": "2016-08-17T00:30:00Z",
      "toDate": "2016-08-23T08:00:00Z",
      "lines": [
        {
          "transport_type": "bus",
          "route_type": 2,
          "line_id": 8118,
          "line_name": "234 - Garden City - City (Queen Victoria Market)",
          "line_number": "234",
          "line_name_short": "Garden City - City (Queen Victoria Market)",
          "line_number_long": "234",
          "direction": null
        },
        {
          "transport_type": "bus",
          "route_type": 2,
          "line_id": 8263,
          "line_name": "236 - Garden City - Queen Victoria Market via City",
          "line_number": "236",
          "line_name_short": "Garden City - Queen Victoria Market via City",
          "line_number_long": "236",
          "direction": null
        }
      ]
    }
  ]
}
```

Stops Nearby

Sample URL

<http://timetableapi.ptv.vic.gov.au/v2/nearme/latitude/-37.81835/longitude/144.966958?devid=1000411&signature=058358CA7063C123F7CAF5CA099A2B74EEA6DF83>

Sample Response

```
[
  {
    "result": {
      "distance": 1.76078174E-09,
      "suburb": "Melbourne City",
      "transport_type": "train",
      "route_type": 0,
      "stop_id": 1071,
      "location_name": "Flinders Street ",
      "lat": -37.81831,
      "lon": 144.966965
    },
    "type": "stop"
  }
]
```

Stopping Pattern

Sample URL

http://timetableapi.ptv.vic.gov.au/v2/mode/0/run/590/stop/1071/stopping-pattern?for_utc=2016-08-19T12:03:29Z&devid=1000411&signature=76B1D919D541FA7298C6C660BED2EC970E2D239E

Sample Response

```
{
  "values": [
    {
      "platform": {
        "realtime_id": 0,
        "stop": {
          "distance": 0.0,
          "suburb": "Melbourne City",
          "transport_type": "train",
          "route_type": 0,
          "stop_id": 1071,
          "location_name": "Flinders Street",
          "lat": -37.81831,
          "lon": 144.966965
        },
        "direction": {
          "linedir_id": 34,
          "direction_id": 0,
          "direction_name": "Alamein",
          "line": {
            "transport_type": "train",
            "route_type": 0,
            "line_id": 1,
            "line_name": "Alamein",
            "line_number": "Alamein",
            "line_name_short": "Alamein",
            "line_number_long": ""
          }
        }
      },
      "run": {
        "transport_type": "train",
        "route_type": 0,
        "run_id": 590,
        "num_skipped": 0,
        "destination_id": 0,
        "destination_name": ""
      },
      "time_timetable_utc": "2016-08-19T12:05:00Z",
      "time_realtime_utc": null,
      "flags": ""
    }
  ]
}
```

Search

Sample URL

[http://timetableapi.ptv.vic.gov.au/v2/search/Sunbury?
devid=1000411&signature=9EB9AE2F1A045548402D619360B2F87C4BFC06E5](http://timetableapi.ptv.vic.gov.au/v2/search/Sunbury?devid=1000411&signature=9EB9AE2F1A045548402D619360B2F87C4BFC06E5)

Sample Response

```
[
  {
    "result": {
      "distance": 0.0,
      "suburb": "Sunbury",
      "transport_type": "bus",
      "route_type": 2,
      "stop_id": 30120,
      "location_name": "115 Gap Rd ",
      "lat": -37.57879,
      "lon": 144.713
    },
    "type": "stop"
  },
  {
    "result": {
      "transport_type": "train",
      "route_type": 0,
      "line_id": 14,
      "line_name": "Sunbury Line",
      "line_number": "Sunbury Line",
      "line_name_short": "Sunbury Line",
      "line_number_long": ""
    },
    "type": "line"
  }
]
```

Stops on a Line

Sample URL

[http://timetableapi.ptv.vic.gov.au/v2/mode/0/line/14/stops-for-line?
devid=1000411&signature=61B2669AD198080F85BCFB435BC5E1A92739F7A3](http://timetableapi.ptv.vic.gov.au/v2/mode/0/line/14/stops-for-line?devid=1000411&signature=61B2669AD198080F85BCFB435BC5E1A92739F7A3)

Sample Response

```
[
  {
    "distance": 0.0,
    "suburb": "Sunshine North",
    "transport_type": "train",
    "route_type": 0,
    "stop_id": 1003,
    "location_name": "Albion Station",
    "lat": -37.7776566,
    "lon": 144.8247
  },
  {
    "distance": 0.0,
    "suburb": "Diggers Rest",
    "transport_type": "train",
    "route_type": 0,
    "stop_id": 1055,
    "location_name": "Diggers Rest Station",
    "lat": -37.6270142,
    "lon": 144.719925
  }
]
```

Next Departures

Sample URL

<http://timetableapi.ptv.vic.gov.au/v2/mode/0/stop/1071/departures/by-destination/limit/5?devid=1000411&signature=893986AD52473FFFBF6A010F8BC4771B843302BD>

Sample Response

```
{
  "values": [
    {
      "platform": {
        "realtime_id": 0,
        "stop": {
          "distance": 0.0,
          "suburb": "Melbourne City",
          "transport_type": "train",
          "route_type": 0,
          "stop_id": 1071,
          "location_name": "Flinders Street",
          "lat": -37.81831,
          "lon": 144.966965
        },
        "direction": {
          "linedir_id": 37,
          "direction_id": 4,
          "direction_name": "Cranbourne",
          "line": {
            "transport_type": "train",
            "route_type": 0,
            "line_id": 4,
            "line_name": "Cranbourne",
            "line_number": "Cranbourne",
            "line_name_short": "Cranbourne",
            "line_number_long": ""
          }
        }
      },
      "run": {
        "transport_type": "train",
        "route_type": 0,
        "run_id": 7909,
        "num_skipped": 0,
        "destination_id": 1049,
        "destination_name": "Dandenong"
      },
      "time_timetable_utc": "2016-08-19T12:03:00Z",
      "time_realtime_utc": null,
      "flags": "",
      "disruptions": [
        {
          "disruption_id": 68893,
          "title": "Cranbourne and Pakenham lines: Buses replacing trains from Saturday 20 August to Sunday, 21 August 2016",
          "url": "http://ptv.vic.gov.au/live-travel-updates/article/cranbourne-and-pakenham-lines-buses-replacing-trains-from-saturday-20-august-to-sunday-21-august-2016",
          "description": "Due to works taking place as part of the Level Crossing Removal Project: Caulfield to Dandenong, buses will replace train services between Caulfield and Westall stations from approximately 12.45am on Saturday 20 August until the last service on Sunday, 21 August 2016.",
          "status": "Planned",
          "type": "Planned Works",
          "publishedOn": "2016-08-11T03:50:15Z",
          "lastUpdated": "2016-08-16T04:17:28Z",
          "fromDate": "2016-08-19T14:00:00Z",
          "toDate": "2016-08-21T17:00:00Z",
          "service_time": null
        },
        {
          "disruption_id": 68491,
          "title": "Planned cancellations on selected lines: Friday afternoons between Friday 5 August and Friday, 26 August 2016",
          "url": "http://ptv.vic.gov.au/live-travel-updates/article/planned-cancellations-on-selected-lines-friday-12-august-2016",
          "description": "Due to driver training requirements, the following planned cancellations will take place on Friday afternoons between Friday 5 August and Friday, 26 August 2016:",
          "status": "Current",
          "type": "Service Information",
          "publishedOn": "2016-08-07T10:09:32Z",
          "lastUpdated": "2016-08-19T04:30:03Z",
          "fromDate": "2016-08-19T04:30:00Z",
          "toDate": "2016-08-26T17:00:00Z",
          "service_time": null
        }
      ]
    }
  ]
}
```

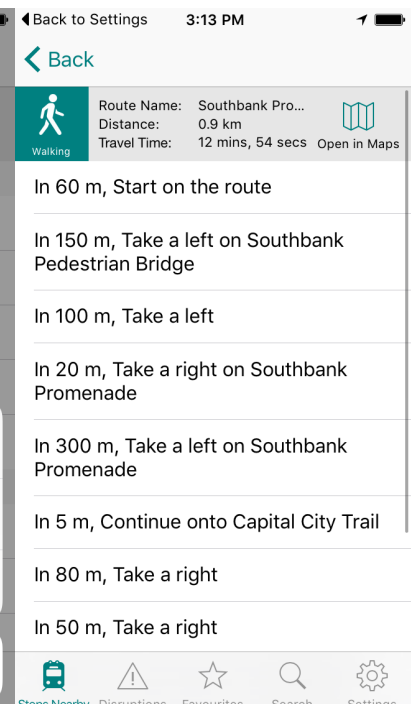
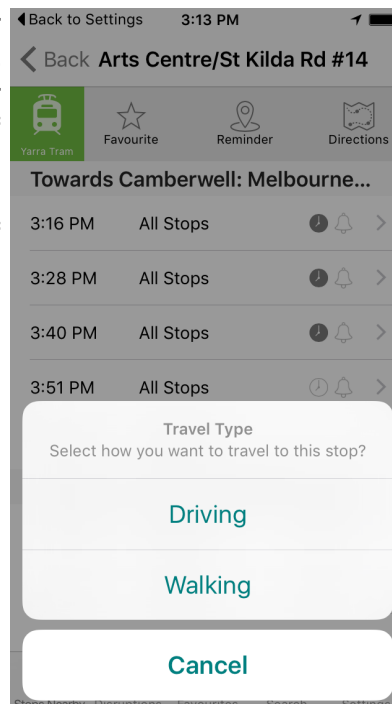
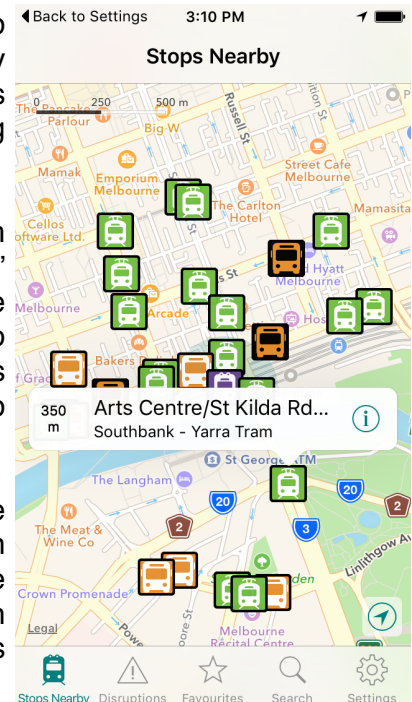

Cocoa Framework Implementation

Trakx implements the MapKit framework in the Stops Nearby tab and its descendant views. Primarily MapKit is used to display custom MKAnnotation's on a MKMapView showing either the user's current location or a pin dropped by the user tapping and holding on the screen.

In order to display the user's location, authorisation from CoreLocation is sought the first time the user taps the "locate" button shown on the bottom right hand side of the map view. If the user had denied access to location services, an alert shows to display available options to the user and the locate button is hidden. This alert also shows the first time the user loads the app after it has been inactive long enough to be removed from memory.

Only one of either user location or dropped pin is visible at any one time as distances displayed on the stop detail view are based on whatever is visible at the time. This distance is displayed in the popup that appears when a stop is tapped, formatted using an MKDistanceFormatter to display appropriate units (i.e. 1200 metres is shown as 1.2 km).

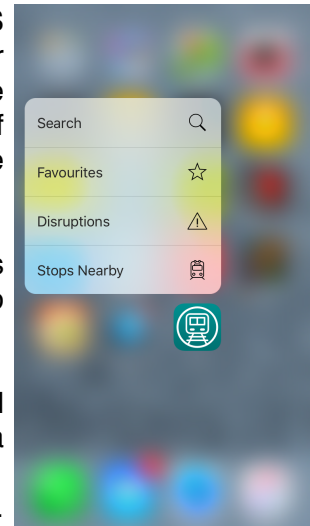
MapKit is also used to provide either written walking or driving directions from the source location (user or dropped pin) to the selected stop. If the user chooses to they can then have the directions displayed using the external Apple Maps or (if installed) Google Maps app.



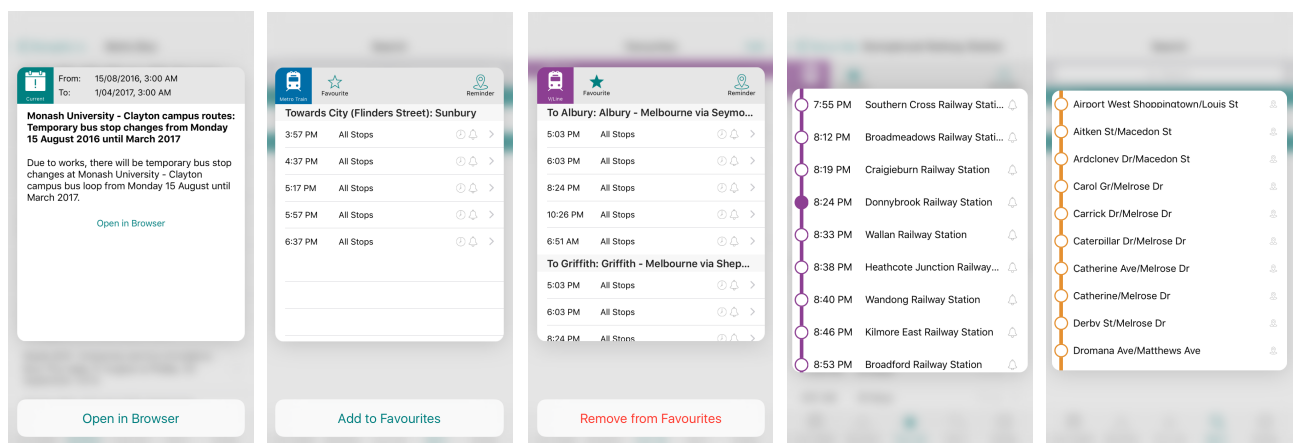
Advanced Feature

Trakx implements 3D touch on devices that support it (currently iPhone 6S and iPhone 6S Plus) in a number of locations. Firstly each of the tab bar options are selectable as menu items by force-touching the app icon on the home screen as shown in the screenshot to the right. Tapping any one of these items will shortcut the user to displaying the appropriate tab in the application immediately on launch.

Trakx also implements 3D touch in the following table views, as demonstrated in the screenshots below. All 3D touch actions are also usable in both portrait and landscape views.



- **Disruptions Table View:** Allows the user to preview the selected disruption. The user can also choose to open the disruption in a browser by swiping up to display the “Open in Browser” action button.
- **Favourites Table View:** Allows the user to preview the selected item. While previewing the user can also remove the favourite by swiping upwards to display a “delete” action button.
- **Search Table View:** Allows the user to preview the selected item. The user can either add or remove from favourites (as applicable) any Stop result.
- **Stop Detail Table View:** Allows the user to preview the selected timetable item.

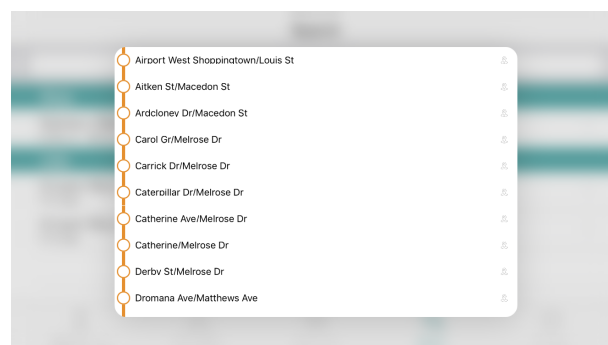


Disruption Preview

Stop Detail Preview
(Add to Favourites)Stop Detail Preview
(Remove from Favourites)

Timetable Preview

Line Detail Preview

Line Detail Preview
(Landscape)

Size Classes

Size Classes have been applied in a number of views across the application, and take into account various size iOS device screens from iPhone 4 up to iPad Pro. The application has been physically tested using iPhone 6 Plus, iPhone 6s and iPad 2 devices, as well as various device sizes on the iOS simulator.

Size classes has been implemented on the following views:

- **Disruptions:** When in a compact height size class, the table view cell text on the Disruption Item List Table View, and Disruption Detail Table View controllers will reduce in size from 16.0 points to 12.0 points to allow more disruptions to be displayed on screen.
- **Search:** As with the disruptions views, when in a compact height size class the primary cell text will reduce in size from 16.0 to 12.0 points, and the sub detail label will change from 11.0 to 9.0 points.
- **Directions:** As with the disruptions views, when in a compact height size class the cell text will reduce in size from 16.0 to 12.0 points.
- **Stop Detail:** When in a compact height size class the table view cell text will reduce in size from 16.0 to 12.0 points. The width of the label containing the departure time will also reduce in width to prevent excessive white space between the time and stopping frequency. The text size behaviour is the reverse of that in the prototype in order to remain consistent with the behaviour in the disruptions and search table views.
- **Line Detail / Stopping Pattern:** As with the disruptions views, when in a compact height size class the cell text will reduce in size from 16.0 to 12.0 points.