'use strict';

angular.module('BlurAdmin', [

'ngAnimate',

'ui.bootstrap',

'ui.sortable',

'ui.router',

'ngTouch',

'toastr',

'smart-table',

"xeditable",

'ui.slimscroll',

'ngJsTree',

'angular-progress-button-styles',

'BlurAdmin.theme',

'BlurAdmin.pages'

]);

'use strict';

var path = require('path');

var gulp = require('gulp');

var conf = require('./conf');

var $ = require('gulp-load-plugins')({

pattern: ['gulp-\*', 'main-bower-files', 'uglify-save-license', 'del']

});

gulp.task('partials', function () {

return gulp.src([

path.join(conf.paths.src, '/app/\*\*/\*.html'),

path.join(conf.paths.tmp, '/serve/app/\*\*/\*.html')

])

.pipe($.minifyHtml({

empty: true,

spare: true,

quotes: true

}))

.pipe($.angularTemplatecache('templateCacheHtml.js', {

module: 'BlurAdmin',

root: 'app'

}))

.pipe(gulp.dest(conf.paths.tmp + '/partials/'));

});

gulp.task('html', ['inject', 'partials'], function () {

var partialsInjectFile = gulp.src(path.join(conf.paths.tmp, '/partials/templateCacheHtml.js'), { read: false });

var partialsInjectOptions = {

starttag: '<!-- inject:partials -->',

ignorePath: path.join(conf.paths.tmp, '/partials'),

addRootSlash: false

};

var htmlFilter = $.filter('\*.html', { restore: true, dot:true});

var jsFilter = $.filter('\*\*/\*.js', { restore: true, dot:true});

var cssFilter = $.filter('\*\*/\*.css', { restore: true, dot:true});

var assets;

return gulp.src(path.join(conf.paths.tmp, '/serve/\*.html'))

.pipe($.inject(partialsInjectFile, partialsInjectOptions))

.pipe(assets = $.useref.assets())

.pipe($.rev())

.pipe(jsFilter)

.pipe($.sourcemaps.init())

.pipe($.ngAnnotate())

.pipe($.uglify({ preserveComments: $.uglifySaveLicense })).on('error', conf.errorHandler('Uglify'))

.pipe($.sourcemaps.write('maps'))

.pipe(jsFilter.restore)

.pipe(cssFilter)

.pipe($.sourcemaps.init())

.pipe($.replace('../../bower\_components/bootstrap-sass/assets/fonts/bootstrap/', '../fonts/'))

.pipe($.minifyCss({ processImport: false }))

.pipe($.sourcemaps.write('maps'))

.pipe(cssFilter.restore)

.pipe(assets.restore())

.pipe($.useref())

.pipe($.revReplace())

.pipe(htmlFilter)

.pipe($.minifyHtml({

empty: true,

spare: true,

quotes: true,

conditionals: true

}))

.pipe(htmlFilter.restore)

.pipe(gulp.dest(path.join(conf.paths.dist, '/')))

.pipe($.size({ title: path.join(conf.paths.dist, '/'), showFiles: true }));

});

// Only applies for fonts from bower dependencies

// Custom fonts are handled by the "other" task

gulp.task('fonts', function () {

return gulp.src($.mainBowerFiles('\*\*/\*.{eot,svg,ttf,woff,woff2}'))

.pipe($.flatten())

.pipe(gulp.dest(path.join(conf.paths.dist, '/fonts/')));

});

gulp.task('other', ['copyVendorImages'], function () {

var fileFilter = $.filter(function (file) {

return file.stat.isFile();

});

return gulp.src([

path.join(conf.paths.src, '/\*\*/\*'),

path.join('!' + conf.paths.src, '/\*\*/\*.{html,css,js,scss,md}'),

path.join(conf.paths.tmp, '/serve/\*\*/assets/img/theme/vendor/\*\*/\*')

])

.pipe(fileFilter)

.pipe(gulp.dest(path.join(conf.paths.dist, '/')));

});

gulp.task('clean', function () {

return $.del([path.join(conf.paths.dist, '/'), path.join(conf.paths.tmp, '/')]);

});

gulp.task('build', ['html', 'fonts', 'other']);

/\*\*

\* This file contains the variables used in other gulp files

\* which defines tasks

\* By design, we only put there very generic config values

\* which are used in several places to keep good readability

\* of the tasks

\*/

var gutil = require('gulp-util');

/\*\*

\* The main paths of your project handle these with care

\*/

exports.paths = {

src: 'src',

dist: 'release',

devDist: 'dev-release',

tmp: '.tmp',

e2e: 'e2e'

};

/\*\*

\* Wiredep is the lib which inject bower dependencies in your project

\* Mainly used to inject script tags in the index.html but also used

\* to inject css preprocessor deps and js files in karma

\*/

exports.wiredep = {

exclude: [/\/bootstrap\.js$/, /\/bootstrap-sass\/.\*\.js/, /\/require\.js/],

directory: 'bower\_components'

};

/\*\*

\* Common implementation for an error handler of a Gulp plugin

\*/

exports.errorHandler = function(title) {

'use strict';

return function(err) {

gutil.log(gutil.colors.red('[' + title + ']'), err.toString());

this.emit('end');

};

};

'use strict';

var path = require('path');

var gulp = require('gulp');

var conf = require('./conf');

var $ = require('gulp-load-plugins')({

pattern: ['gulp-\*', 'main-bower-files']

});

var \_ = require('lodash');

gulp.task('dev-fonts', function () {

return gulp.src($.mainBowerFiles())

.pipe($.filter('\*\*/\*.{eot,svg,ttf,woff,woff2}'))

.pipe($.flatten())

.pipe(gulp.dest(path.join(conf.paths.devDist, 'fonts')));

});

gulp.task('dev-copy-lib', function () {

var assets = require('wiredep')(\_.extend({}, conf.wiredep));

var srcList = [];

srcList.push.apply(srcList, assets.js);

srcList.push.apply(srcList, assets.css);

return gulp

.src(srcList/\*, { base: '.' }\*/)

/\* .pipe($.rename(function (p) {

p.dirname = p.dirname.replace(/\\/g, '/').replace('bower\_components/', '');

if (p.dirname.indexOf('/') !== -1) {

p.dirname = p.dirname.substr(0, p.dirname.indexOf('/'));

}

}))\*/

.pipe(gulp.dest(path.join(conf.paths.devDist, 'lib')));

});

gulp.task('dev-css-replace', ['dev-copy-assets'], function() {

return gulp.src(path.join(conf.paths.devDist, '\*.html'))

.pipe($.replace(/<link rel="stylesheet" href="\.\.\/bower\_components\/.\*\/(.\*)"\s\*?\/>/g, '<link rel="stylesheet" href="lib/$1" >'))

.pipe(gulp.dest(conf.paths.devDist));

});

gulp.task('dev-js-replace', ['dev-copy-assets'], function() {

return gulp.src(path.join(conf.paths.devDist, '.html'))

.pipe($.replace(/<script src="\.\.\/bower\_components\/.\*\/(.\*)"\s\*?>/g, '<script src="lib/$1">'))

.pipe(gulp.dest(conf.paths.devDist));

});

gulp.task('dev-copy-assets', ['inject', 'dev-copy-lib', 'dev-fonts'], function () {

return gulp

.src([

conf.paths.src + '/\*\*/\*',

path.join(conf.paths.tmp, '/serve/\*\*/\*')

])

.pipe(gulp.dest(conf.paths.devDist));

});

gulp.task('dev-release', ['dev-css-replace', 'dev-js-replace']);

'use strict';

var gulp = require('gulp');

var $ = require('gulp-load-plugins')();

gulp.task('wintersmith-generate', $.shell.task([

'wintersmith build'

], { cwd: 'docs' }));

gulp.task('deploy-docs', ['wintersmith-generate'], function() {

return gulp.src('./docs/build/\*\*/\*')

.pipe($.ghPages());

});

'use strict';

var path = require('path');

var gulp = require('gulp');

var conf = require('./conf');

gulp.task('copyVendorImages', function () {

return gulp

.src([

path.join(conf.wiredep.directory, '\*\*/ammap/dist/ammap/images/\*\*/\*'),

path.join(conf.wiredep.directory, '\*\*/amcharts/dist/amcharts/images/\*\*/\*'),

path.join(conf.wiredep.directory, '\*\*/ionrangeslider/img/\*\*/\*'),

path.join(conf.wiredep.directory, '\*\*/jstree/dist/themes/\*\*/\*'),

path.join(conf.wiredep.directory, '\*\*/leaflet/dist/images/\*\*/\*')

])

.pipe(gulp.dest(path.join(conf.paths.tmp, 'serve', '/assets/img/theme/vendor')));

});

'use strict';

var path = require('path');

var gulp = require('gulp');

var conf = require('./conf');

var $ = require('gulp-load-plugins')();

var wiredep = require('wiredep').stream;

var \_ = require('lodash');

var browserSync = require('browser-sync');

gulp.task('inject-reload', ['inject'], function () {

browserSync.reload();

});

gulp.task('inject', ['scripts', 'styles', 'injectAuth', 'inject404', 'copyVendorImages'], function () {

var injectStyles = gulp.src([

path.join(conf.paths.tmp, '/serve/app/main.css'),

path.join('!' + conf.paths.tmp, '/serve/app/vendor.css')

], {read: false});

var injectScripts = gulp.src([

path.join(conf.paths.src, '/assets/js/\*\*/\*.js'),

path.join(conf.paths.src, '/app/\*\*/\*.module.js'),

path.join(conf.paths.src, '/app/\*\*/\*.js'),

path.join('!' + conf.paths.src, '/app/\*\*/\*.spec.js'),

path.join('!' + conf.paths.src, '/app/\*\*/\*.mock.js'),

])

/\*.pipe($.angularFilesort())\*/.on('error', conf.errorHandler('AngularFilesort'));

var injectOptions = {

ignorePath: [conf.paths.src, path.join(conf.paths.tmp, '/serve')],

addRootSlash: false

};

return gulp.src(path.join(conf.paths.src, '/index.html'))

.pipe($.inject(injectStyles, injectOptions))

.pipe($.inject(injectScripts, injectOptions))

.pipe(wiredep(\_.extend({}, conf.wiredep)))

.pipe(gulp.dest(path.join(conf.paths.tmp, '/serve')));

});

gulp.task('injectAuth', ['stylesAuth'], function () {

return injectAlone({

css: [path.join('!' + conf.paths.tmp, '/serve/app/vendor.css'), path.join(conf.paths.tmp, '/serve/app/auth.css')],

paths: [path.join(conf.paths.src, '/auth.html'), path.join(conf.paths.src, '/reg.html')]

})

});

gulp.task('inject404', ['styles404'], function () {

return injectAlone({

css: [path.join('!' + conf.paths.tmp, '/serve/app/vendor.css'), path.join(conf.paths.tmp, '/serve/app/404.css')],

paths: path.join(conf.paths.src, '/404.html')

})

});

var injectAlone = function (options) {

var injectStyles = gulp.src(

options.css

, {read: false});

var injectOptions = {

ignorePath: [conf.paths.src, path.join(conf.paths.tmp, '/serve')],

addRootSlash: false

};

return gulp.src(options.paths)

.pipe($.inject(injectStyles, injectOptions))

.pipe(wiredep(\_.extend({}, conf.wiredep)))

.pipe(gulp.dest(path.join(conf.paths.tmp, '/serve')));

};

'use strict';

var path = require('path');

var gulp = require('gulp');

var zip = require('gulp-zip');

var prompt = require('gulp-prompt');

var rename = require('gulp-rename');

gulp.task('marketplace-release', ['build', 'dev-release'], function () {

return gulp.src('')

.pipe(prompt.prompt({

type: 'input',

name: 'version',

message: 'Please enter release version (x.x.x)'

}, function (res) {

var nameAndVersion = 'blur-admin-' + res.version;

return gulp

.src(['src/\*\*', 'release/\*\*', 'dev-release/\*\*', 'gulp/\*\*', 'bower.json', 'gulpfile.js', 'package.json', 'README.md', '.gitignore'], {base: "."})

.pipe(rename(function (path) {

path.dirname = nameAndVersion + '/' + path.dirname;

}))

.pipe(zip(nameAndVersion + '.zip'))

.pipe(gulp.dest('.'));

}));

});

'use strict';

var path = require('path');

var gulp = require('gulp');

var conf = require('./conf');

var browserSync = require('browser-sync');

var $ = require('gulp-load-plugins')();

gulp.task('scripts-reload', function() {

return buildScripts()

.pipe(browserSync.stream());

});

gulp.task('scripts', function() {

return buildScripts();

});

function buildScripts() {

return gulp.src(path.join(conf.paths.src, '/app/\*\*/\*.js'))

.pipe($.eslint())

.pipe($.eslint.format())

.pipe($.size())

};

'use strict';

var path = require('path');

var gulp = require('gulp');

var conf = require('./conf');

var browserSync = require('browser-sync');

var browserSyncSpa = require('browser-sync-spa');

var util = require('util');

var proxyMiddleware = require('http-proxy-middleware');

function browserSyncInit(baseDir, browser) {

browser = browser === undefined ? 'default' : browser;

var routes = null;

if(baseDir === conf.paths.src || (util.isArray(baseDir) && baseDir.indexOf(conf.paths.src) !== -1)) {

routes = {

'/bower\_components': 'bower\_components'

};

}

var server = {

baseDir: baseDir,

routes: routes

};

/\*

\* You can add a proxy to your backend by uncommenting the line below.

\* You just have to configure a context which will we redirected and the target url.

\* Example: $http.get('/users') requests will be automatically proxified.

\*

\* For more details and option, https://github.com/chimurai/http-proxy-middleware/blob/v0.9.0/README.md

\*/

// server.middleware = proxyMiddleware('/users', {target: 'http://jsonplaceholder.typicode.com', changeOrigin: true});

browserSync.instance = browserSync.init({

startPath: '/',

server: server,

browser: browser,

ghostMode: false

});

}

browserSync.use(browserSyncSpa({

selector: '[ng-app]'// Only needed for angular apps

}));

gulp.task('serve', ['watch'], function () {

browserSyncInit([path.join(conf.paths.tmp, '/serve'), conf.paths.src]);

});

gulp.task('serve:dist', ['build'], function () {

browserSyncInit(conf.paths.dist);

});

gulp.task('serve:e2e', ['inject'], function () {

browserSyncInit([conf.paths.tmp + '/serve', conf.paths.src], []);

});

gulp.task('serve:e2e-dist', ['build'], function () {

browserSyncInit(conf.paths.dist, []);

});

'use strict';

var path = require('path');

var gulp = require('gulp');

var conf = require('./conf');

var browserSync = require('browser-sync');

var $ = require('gulp-load-plugins')();

var wiredep = require('wiredep').stream;

var \_ = require('lodash');

gulp.task('styles-reload', ['styles'], function () {

return buildStyles()

.pipe(browserSync.stream());

});

gulp.task('styles', function () {

return buildStyles();

});

gulp.task('stylesAuth', function () {

return buildSingleScss(path.join(conf.paths.src, '/sass/auth.scss'));

});

gulp.task('styles404', function () {

return buildSingleScss(path.join(conf.paths.src, '/sass/404.scss'));

});

var buildStyles = function () {

var sassOptions = {

style: 'expanded'

};

var injectFiles = gulp.src([

path.join(conf.paths.src, '/sass/\*\*/\_\*.scss'),

'!' + path.join(conf.paths.src, '/sass/theme/conf/\*\*/\*.scss'),

'!' + path.join(conf.paths.src, '/sass/404.scss'),

'!' + path.join(conf.paths.src, '/sass/auth.scss')

], {read: false});

var injectOptions = {

transform: function (filePath) {

filePath = filePath.replace(conf.paths.src + '/sass/', '');

return '@import "' + filePath + '";';

},

starttag: '// injector',

endtag: '// endinjector',

addRootSlash: false

};

return gulp.src([

path.join(conf.paths.src, '/sass/main.scss')

])

.pipe($.inject(injectFiles, injectOptions))

.pipe(wiredep(\_.extend({}, conf.wiredep)))

.pipe($.sourcemaps.init())

.pipe($.sass(sassOptions)).on('error', conf.errorHandler('Sass'))

.pipe($.autoprefixer()).on('error', conf.errorHandler('Autoprefixer'))

.pipe($.sourcemaps.write())

.pipe(gulp.dest(path.join(conf.paths.tmp, '/serve/app/')));

};

var buildSingleScss = function (paths) {

var sassOptions = {

style: 'expanded'

};

return gulp.src([paths])

.pipe($.sass(sassOptions)).on('error', conf.errorHandler('Sass'))

.pipe($.autoprefixer()).on('error', conf.errorHandler('Autoprefixer'))

.pipe(gulp.dest(path.join(conf.paths.tmp, '/serve/app/')));

};

'use strict';

var path = require('path');

var gulp = require('gulp');

var conf = require('./conf');

var browserSync = require('browser-sync');

function isOnlyChange(event) {

return event.type === 'changed';

}

gulp.task('watch', ['inject'], function () {

gulp.watch([path.join(conf.paths.src, '/\*.html'), 'bower.json'], ['inject-reload']);

gulp.watch([

path.join(conf.paths.src, '/sass/\*\*/\*.css'),

path.join(conf.paths.src, '/sass/\*\*/\*.scss')

], function(event) {

if(isOnlyChange(event)) {

gulp.start('styles-reload');

} else {

gulp.start('inject-reload');

}

});

gulp.watch(path.join(conf.paths.src, '/app/\*\*/\*.js'), function(event) {

if(isOnlyChange(event)) {

gulp.start('scripts-reload');

} else {

gulp.start('inject-reload');

}

});

gulp.watch(path.join(conf.paths.src, '/app/\*\*/\*.html'), function(event) {

browserSync.reload(event.path);

});

});

(function () {

var d;

window.AmCharts ? d = window.AmCharts : (d = {}, window.AmCharts = d, d.themes = {}, d.maps = {}, d.inheriting = {}, d.charts = [], d.onReadyArray = [], d.useUTC = !1, d.updateRate = 30, d.uid = 0, d.lang = {}, d.translations = {}, d.mapTranslations = {}, d.windows = {}, d.initHandlers = []);

d.Class = function (a) {

var b = function () {

arguments[0] !== d.inheriting && (this.events = {}, this.construct.apply(this, arguments))

};

a.inherits ? (b.prototype = new a.inherits(d.inheriting), b.base = a.inherits.prototype, delete a.inherits) : (b.prototype.createEvents =

function () {

for (var a = 0, b = arguments.length; a < b; a++) this.events[arguments[a]] = []

}, b.prototype.listenTo = function (a, b, c) {

this.removeListener(a, b, c);

a.events[b].push({handler: c, scope: this})

}, b.prototype.addListener = function (a, b, c) {

this.removeListener(this, a, b);

this.events[a].push({handler: b, scope: c})

}, b.prototype.removeListener = function (a, b, c) {

if (a && a.events) for (a = a.events[b], b = a.length - 1; 0 <= b; b--) a[b].handler === c && a.splice(b, 1)

}, b.prototype.fire = function (a, b) {

for (var c = this.events[a], d = 0, k = c.length; d <

k; d++) {

var l = c[d];

l.handler.call(l.scope, b)

}

});

for (var c in a) b.prototype[c] = a[c];

return b

};

d.addChart = function (a) {

d.updateInt || (d.updateInt = setInterval(function () {

d.update()

}, Math.round(1E3 / d.updateRate)));

d.charts.push(a)

};

d.removeChart = function (a) {

for (var b = d.charts, c = b.length - 1; 0 <= c; c--) b[c] == a && b.splice(c, 1);

0 === b.length && d.updateInt && (clearInterval(d.updateInt), d.updateInt = NaN)

};

d.isModern = !0;

d.getIEVersion = function () {

var a = 0, b, c;

"Microsoft Internet Explorer" == navigator.appName && (b = navigator.userAgent,

c = /MSIE ([0-9]{1,}[.0-9]{0,})/, null !== c.exec(b) && (a = parseFloat(RegExp.$1)));

return a

};

d.applyLang = function (a, b) {

var c = d.translations;

b.dayNames = d.extend({}, d.dayNames);

b.shortDayNames = d.extend({}, d.shortDayNames);

b.monthNames = d.extend({}, d.monthNames);

b.shortMonthNames = d.extend({}, d.shortMonthNames);

c && (c = c[a]) && (d.lang = c, c.monthNames && (b.dayNames = d.extend({}, c.dayNames), b.shortDayNames = d.extend({}, c.shortDayNames), b.monthNames = d.extend({}, c.monthNames), b.shortMonthNames = d.extend({}, c.shortMonthNames)))

};

d.IEversion = d.getIEVersion();

9 > d.IEversion && 0 < d.IEversion && (d.isModern = !1, d.isIE = !0);

d.dx = 0;

d.dy = 0;

if (document.addEventListener || window.opera) d.isNN = !0, d.isIE = !1, d.dx = .5, d.dy = .5;

document.attachEvent && (d.isNN = !1, d.isIE = !0, d.isModern || (d.dx = 0, d.dy = 0));

window.chrome && (d.chrome = !0);

d.handleMouseUp = function (a) {

for (var b = d.charts, c = 0; c < b.length; c++) {

var e = b[c];

e && e.handleReleaseOutside && e.handleReleaseOutside(a)

}

};

d.handleMouseMove = function (a) {

for (var b = d.charts, c = 0; c < b.length; c++) {

var e = b[c];

e && e.handleMouseMove &&

e.handleMouseMove(a)

}

};

d.handleWheel = function (a) {

for (var b = d.charts, c = 0; c < b.length; c++) {

var e = b[c];

if (e && e.mouseIsOver) {

e.mouseWheelScrollEnabled || e.mouseWheelZoomEnabled ? e.handleWheel && e.handleWheel(a) : a.stopPropagation && a.stopPropagation();

break

}

}

};

d.resetMouseOver = function () {

for (var a = d.charts, b = 0; b < a.length; b++) {

var c = a[b];

c && (c.mouseIsOver = !1)

}

};

d.ready = function (a) {

d.onReadyArray.push(a)

};

d.handleLoad = function () {

d.isReady = !0;

for (var a = d.onReadyArray, b = 0; b < a.length; b++) {

var c = a[b];

isNaN(d.processDelay) ?

c() : setTimeout(c, d.processDelay \* b)

}

};

d.addInitHandler = function (a, b) {

d.initHandlers.push({method: a, types: b})

};

d.callInitHandler = function (a) {

var b = d.initHandlers;

if (d.initHandlers) for (var c = 0; c < b.length; c++) {

var e = b[c];

e.types ? d.isInArray(e.types, a.type) && e.method(a) : e.method(a)

}

};

d.getUniqueId = function () {

d.uid++;

return "AmChartsEl-" + d.uid

};

d.isNN && (document.addEventListener("mousemove", d.handleMouseMove, !0), document.addEventListener("mouseup", d.handleMouseUp, !0), window.addEventListener("load", d.handleLoad,

!0), window.addEventListener("DOMMouseScroll", d.handleWheel, !0), document.addEventListener("mousewheel", d.handleWheel, !0));

d.isIE && (document.attachEvent("onmousemove", d.handleMouseMove), document.attachEvent("onmouseup", d.handleMouseUp), window.attachEvent("onload", d.handleLoad));

d.clear = function () {

var a = d.charts;

if (a) for (var b = a.length - 1; 0 <= b; b--) a[b].clear();

d.updateInt && clearInterval(d.updateInt);

d.charts = [];

d.isNN && (document.removeEventListener("mousemove", d.handleMouseMove, !0), document.removeEventListener("mouseup",

d.handleMouseUp, !0), window.removeEventListener("load", d.handleLoad, !0), window.removeEventListener("DOMMouseScroll", d.handleWheel, !0), document.removeEventListener("mousewheel", d.handleWheel, !0));

d.isIE && (document.detachEvent("onmousemove", d.handleMouseMove), document.detachEvent("onmouseup", d.handleMouseUp), window.detachEvent("onload", d.handleLoad))

};

d.makeChart = function (a, b, c) {

var e = b.type, h = b.theme;

d.isString(h) && (h = d.themes[h], b.theme = h);

var f;

switch (e) {

case "serial":

f = new d.AmSerialChart(h);

break;

case "xy":

f = new d.AmXYChart(h);

break;

case "pie":

f = new d.AmPieChart(h);

break;

case "radar":

f = new d.AmRadarChart(h);

break;

case "gauge":

f = new d.AmAngularGauge(h);

break;

case "funnel":

f = new d.AmFunnelChart(h);

break;

case "map":

f = new d.AmMap(h);

break;

case "stock":

f = new d.AmStockChart(h);

break;

case "gantt":

f = new d.AmGanttChart(h)

}

d.extend(f, b);

if (b.listeners) for (var g in b.listeners) e = b.listeners[g], f.addListener(e.event, e.method);

d.isReady ? isNaN(c) ? f.write(a) : setTimeout(function () {

d.realWrite(f, a)

}, c) : d.ready(function () {

isNaN(c) ?

f.write(a) : setTimeout(function () {

d.realWrite(f, a)

}, c)

});

return f

};

d.realWrite = function (a, b) {

a.write(b)

};

d.updateCount = 0;

d.validateAt = Math.round(d.updateRate / 5);

d.update = function () {

var a = d.charts;

d.updateCount++;

var b = !1;

d.updateCount == d.validateAt && (b = !0, d.updateCount = 0);

if (a) for (var c = 0; c < a.length; c++) a[c].update && a[c].update(), b && a[c].autoResize && a[c].validateSize && a[c].validateSize()

};

d.bezierX = 3;

d.bezierY = 6

})();

(function () {

var d = window.AmCharts;

d.toBoolean = function (a, b) {

if (void 0 === a) return b;

switch (String(a).toLowerCase()) {

case "true":

case "yes":

case "1":

return !0;

case "false":

case "no":

case "0":

case null:

return !1;

default:

return Boolean(a)

}

};

d.removeFromArray = function (a, b) {

var c;

if (void 0 !== b && void 0 !== a) for (c = a.length - 1; 0 <= c; c--) a[c] == b && a.splice(c, 1)

};

d.getPath = function () {

var a = document.getElementsByTagName("script");

if (a) for (var b = 0; b < a.length; b++) {

var c = a[b].src;

if (-1 !== c.search(/\/(amcharts|ammap)\.js/)) return c.replace(/\/(amcharts|ammap)\.js.\*/,

"/")

}

};

d.normalizeUrl = function (a) {

return "" !== a && -1 === a.search(/\/$/) ? a + "/" : a

};

d.isAbsolute = function (a) {

return 0 === a.search(/^http[s]?:|^\//)

};

d.isInArray = function (a, b) {

for (var c = 0; c < a.length; c++) if (a[c] == b) return !0;

return !1

};

d.getDecimals = function (a) {

var b = 0;

isNaN(a) || (a = String(a), -1 != a.indexOf("e-") ? b = Number(a.split("-")[1]) : -1 != a.indexOf(".") && (b = a.split(".")[1].length));

return b

};

d.wordwrap = function (a, b, c, e) {

var h, f, g, k;

a += "";

if (1 > b) return a;

h = -1;

for (a = (k = a.split(/\r\n|\n|\r/)).length; ++h < a; k[h] +=

g) {

g = k[h];

for (k[h] = ""; g.length > b; k[h] += d.trim(g.slice(0, f)) + ((g = g.slice(f)).length ? c : "")) f = 2 == e || (f = g.slice(0, b + 1).match(/\S\*(\s)?$/))[1] ? b : f.input.length - f[0].length || 1 == e && b || f.input.length + (f = g.slice(b).match(/^\S\*/))[0].length;

g = d.trim(g)

}

return k.join(e)

};

d.trim = function (a) {

return a.replace(/^[\s\uFEFF\xA0]+|[\s\uFEFF\xA0]+$/g, "")

};

d.wrappedText = function (a, b, c, e, h, f, g, k) {

var l = d.text(a, b, c, e, h, f, g);

if (l) {

var m = l.getBBox();

if (m.width > k) {

var n = "\n";

d.isModern || (n = "<br>");

k = Math.floor(k / (m.width /

b.length));

2 < k && (k -= 2);

b = d.wordwrap(b, k, n, !0);

l.remove();

l = d.text(a, b, c, e, h, f, g)

}

}

return l

};

d.getStyle = function (a, b) {

var c = "";

document.defaultView && document.defaultView.getComputedStyle ? c = document.defaultView.getComputedStyle(a, "").getPropertyValue(b) : a.currentStyle && (b = b.replace(/\-(\w)/g, function (a, b) {

return b.toUpperCase()

}), c = a.currentStyle[b]);

return c

};

d.removePx = function (a) {

if (void 0 !== a) return Number(a.substring(0, a.length - 2))

};

d.getURL = function (a, b) {

if (a) if ("\_self" != b && b) if ("\_top" == b && window.top) window.top.location.href =

a; else if ("\_parent" == b && window.parent) window.parent.location.href = a; else if ("\_blank" == b) window.open(a); else {

var c = document.getElementsByName(b)[0];

c ? c.src = a : (c = d.windows[b]) ? c.opener && !c.opener.closed ? c.location.href = a : d.windows[b] = window.open(a) : d.windows[b] = window.open(a)

} else window.location.href = a

};

d.ifArray = function (a) {

return a && "object" == typeof a && 0 < a.length ? !0 : !1

};

d.callMethod = function (a, b) {

var c;

for (c = 0; c < b.length; c++) {

var e = b[c];

if (e) {

if (e[a]) e[a]();

var h = e.length;

if (0 < h) {

var d;

for (d = 0; d <

h; d++) {

var g = e[d];

if (g && g[a]) g[a]()

}

}

}

}

};

d.toNumber = function (a) {

return "number" == typeof a ? a : Number(String(a).replace(/[^0-9\-.]+/g, ""))

};

d.toColor = function (a) {

if ("" !== a && void 0 !== a) if (-1 != a.indexOf(",")) {

a = a.split(",");

var b;

for (b = 0; b < a.length; b++) {

var c = a[b].substring(a[b].length - 6, a[b].length);

a[b] = "#" + c

}

} else a = a.substring(a.length - 6, a.length), a = "#" + a;

return a

};

d.toCoordinate = function (a, b, c) {

var e;

void 0 !== a && (a = String(a), c && c < b && (b = c), e = Number(a), -1 != a.indexOf("!") && (e = b - Number(a.substr(1))), -1 !=

a.indexOf("%") && (e = b \* Number(a.substr(0, a.length - 1)) / 100));

return e

};

d.fitToBounds = function (a, b, c) {

a < b && (a = b);

a > c && (a = c);

return a

};

d.isDefined = function (a) {

return void 0 === a ? !1 : !0

};

d.stripNumbers = function (a) {

return a.replace(/[0-9]+/g, "")

};

d.roundTo = function (a, b) {

if (0 > b) return a;

var c = Math.pow(10, b);

return Math.round(a \* c) / c

};

d.toFixed = function (a, b) {

var c = String(Math.round(a \* Math.pow(10, b)));

if (0 < b) {

var e = c.length;

if (e < b) {

var h;

for (h = 0; h < b - e; h++) c = "0" + c

}

e = c.substring(0, c.length - b);

"" === e && (e = 0);

return e +

"." + c.substring(c.length - b, c.length)

}

return String(c)

};

d.formatDuration = function (a, b, c, e, h, f) {

var g = d.intervals, k = f.decimalSeparator;

if (a >= g[b].contains) {

var l = a - Math.floor(a / g[b].contains) \* g[b].contains;

"ss" == b ? (l = d.formatNumber(l, f), 1 == l.split(k)[0].length && (l = "0" + l)) : l = d.roundTo(l, f.precision);

("mm" == b || "hh" == b) && 10 > l && (l = "0" + l);

c = l + "" + e[b] + "" + c;

a = Math.floor(a / g[b].contains);

b = g[b].nextInterval;

return d.formatDuration(a, b, c, e, h, f)

}

"ss" == b && (a = d.formatNumber(a, f), 1 == a.split(k)[0].length && (a = "0" +

a));

("mm" == b || "hh" == b) && 10 > a && (a = "0" + a);

c = a + "" + e[b] + "" + c;

if (g[h].count > g[b].count) for (a = g[b].count; a < g[h].count; a++) b = g[b].nextInterval, "ss" == b || "mm" == b || "hh" == b ? c = "00" + e[b] + "" + c : "DD" == b && (c = "0" + e[b] + "" + c);

":" == c.charAt(c.length - 1) && (c = c.substring(0, c.length - 1));

return c

};

d.formatNumber = function (a, b, c, e, h) {

a = d.roundTo(a, b.precision);

isNaN(c) && (c = b.precision);

var f = b.decimalSeparator;

b = b.thousandsSeparator;

var g;

g = 0 > a ? "-" : "";

a = Math.abs(a);

var k = String(a), l = !1;

-1 != k.indexOf("e") && (l = !0);

0 <= c && !l && (k =

d.toFixed(a, c));

var m = "";

if (l) m = k; else {

var k = k.split("."), l = String(k[0]), n;

for (n = l.length; 0 <= n; n -= 3) m = n != l.length ? 0 !== n ? l.substring(n - 3, n) + b + m : l.substring(n - 3, n) + m : l.substring(n - 3, n);

void 0 !== k[1] && (m = m + f + k[1]);

void 0 !== c && 0 < c && "0" != m && (m = d.addZeroes(m, f, c))

}

m = g + m;

"" === g && !0 === e && 0 !== a && (m = "+" + m);

!0 === h && (m += "%");

return m

};

d.addZeroes = function (a, b, c) {

a = a.split(b);

void 0 === a[1] && 0 < c && (a[1] = "0");

return a[1].length < c ? (a[1] += "0", d.addZeroes(a[0] + b + a[1], b, c)) : void 0 !== a[1] ? a[0] + b + a[1] : a[0]

};

d.scientificToNormal =

function (a) {

var b;

a = String(a).split("e");

var c;

if ("-" == a[1].substr(0, 1)) {

b = "0.";

for (c = 0; c < Math.abs(Number(a[1])) - 1; c++) b += "0";

b += a[0].split(".").join("")

} else {

var e = 0;

b = a[0].split(".");

b[1] && (e = b[1].length);

b = a[0].split(".").join("");

for (c = 0; c < Math.abs(Number(a[1])) - e; c++) b += "0"

}

return b

};

d.toScientific = function (a, b) {

if (0 === a) return "0";

var c = Math.floor(Math.log(Math.abs(a)) \* Math.LOG10E), e = String(e).split(".").join(b);

return String(e) + "e" + c

};

d.randomColor = function () {

return "#" + ("00000" + (16777216 \* Math.random() <<

0).toString(16)).substr(-6)

};

d.hitTest = function (a, b, c) {

var e = !1, h = a.x, f = a.x + a.width, g = a.y, k = a.y + a.height, l = d.isInRectangle;

e || (e = l(h, g, b));

e || (e = l(h, k, b));

e || (e = l(f, g, b));

e || (e = l(f, k, b));

e || !0 === c || (e = d.hitTest(b, a, !0));

return e

};

d.isInRectangle = function (a, b, c) {

return a >= c.x - 5 && a <= c.x + c.width + 5 && b >= c.y - 5 && b <= c.y + c.height + 5 ? !0 : !1

};

d.isPercents = function (a) {

if (-1 != String(a).indexOf("%")) return !0

};

d.findPosX = function (a) {

var b = a, c = a.offsetLeft;

if (a.offsetParent) {

for (; a = a.offsetParent;) c += a.offsetLeft;

for (; (b = b.parentNode) && b != document.body;) c -= b.scrollLeft || 0

}

return c

};

d.findPosY = function (a) {

var b = a, c = a.offsetTop;

if (a.offsetParent) {

for (; a = a.offsetParent;) c += a.offsetTop;

for (; (b = b.parentNode) && b != document.body;) c -= b.scrollTop || 0

}

return c

};

d.findIfFixed = function (a) {

if (a.offsetParent) for (; a = a.offsetParent;) if ("fixed" == d.getStyle(a, "position")) return !0;

return !1

};

d.findIfAuto = function (a) {

return a.style && "auto" == d.getStyle(a, "overflow") ? !0 : a.parentNode ? d.findIfAuto(a.parentNode) : !1

};

d.findScrollLeft =

function (a, b) {

a.scrollLeft && (b += a.scrollLeft);

return a.parentNode ? d.findScrollLeft(a.parentNode, b) : b

};

d.findScrollTop = function (a, b) {

a.scrollTop && (b += a.scrollTop);

return a.parentNode ? d.findScrollTop(a.parentNode, b) : b

};

d.formatValue = function (a, b, c, e, h, f, g, k) {

if (b) {

void 0 === h && (h = "");

var l;

for (l = 0; l < c.length; l++) {

var m = c[l], n = b[m];

void 0 !== n && (n = f ? d.addPrefix(n, k, g, e) : d.formatNumber(n, e), a = a.replace(new RegExp("\\[\\[" + h + "" + m + "\\]\\]", "g"), n))

}

}

return a

};

d.formatDataContextValue = function (a, b) {

if (a) {

var c =

a.match(/\[\[.\*?\]\]/g), e;

for (e = 0; e < c.length; e++) {

var h = c[e], h = h.substr(2, h.length - 4);

void 0 !== b[h] && (a = a.replace(new RegExp("\\[\\[" + h + "\\]\\]", "g"), b[h]))

}

}

return a

};

d.massReplace = function (a, b) {

for (var c in b) if (b.hasOwnProperty(c)) {

var e = b[c];

void 0 === e && (e = "");

a = a.replace(c, e)

}

return a

};

d.cleanFromEmpty = function (a) {

return a.replace(/\[\[[^\]]\*\]\]/g, "")

};

d.addPrefix = function (a, b, c, e, h) {

var f = d.formatNumber(a, e), g = "", k, l, m;

if (0 === a) return "0";

0 > a && (g = "-");

a = Math.abs(a);

if (1 < a) for (k = b.length - 1; -1 <

k; k--) {

if (a >= b[k].number && (l = a / b[k].number, m = Number(e.precision), 1 > m && (m = 1), c = d.roundTo(l, m), m = d.formatNumber(c, {

precision: -1,

decimalSeparator: e.decimalSeparator,

thousandsSeparator: e.thousandsSeparator

}), !h || l == c)) {

f = g + "" + m + "" + b[k].prefix;

break

}

} else for (k = 0; k < c.length; k++) if (a <= c[k].number) {

l = a / c[k].number;

m = Math.abs(Math.round(Math.log(l) \* Math.LOG10E));

l = d.roundTo(l, m);

f = g + "" + l + "" + c[k].prefix;

break

}

return f

};

d.remove = function (a) {

a && a.remove()

};

d.getEffect = function (a) {

">" == a && (a = "easeOutSine");

"<" ==

a && (a = "easeInSine");

"elastic" == a && (a = "easeOutElastic");

return a

};

d.getObjById = function (a, b) {

var c, e;

for (e = 0; e < a.length; e++) {

var h = a[e];

h.id == b && (c = h)

}

return c

};

d.applyTheme = function (a, b, c) {

b || (b = d.theme);

b && b[c] && d.extend(a, b[c])

};

d.isString = function (a) {

return "string" == typeof a ? !0 : !1

};

d.extend = function (a, b, c) {

var e;

a || (a = {});

for (e in b) c ? a.hasOwnProperty(e) || (a[e] = b[e]) : a[e] = b[e];

return a

};

d.copyProperties = function (a, b) {

for (var c in a) a.hasOwnProperty(c) && "events" != c && void 0 !== a[c] && "function" != typeof a[c] &&

"cname" != c && (b[c] = a[c])

};

d.processObject = function (a, b, c, e) {

if (!1 === a instanceof b && (a = e ? d.extend(new b(c), a) : d.extend(a, new b(c), !0), a.listeners)) for (var h in a.listeners) b = a.listeners[h], a.addListener(b.event, b.method);

return a

};

d.fixNewLines = function (a) {

var b = RegExp("\\n", "g");

a && (a = a.replace(b, "<br />"));

return a

};

d.fixBrakes = function (a) {

if (d.isModern) {

var b = RegExp("<br>", "g");

a && (a = a.replace(b, "\n"))

} else a = d.fixNewLines(a);

return a

};

d.deleteObject = function (a, b) {

if (a) {

if (void 0 === b || null === b) b =

20;

if (0 !== b) if ("[object Array]" === Object.prototype.toString.call(a)) for (var c = 0; c < a.length; c++) d.deleteObject(a[c], b - 1), a[c] = null; else if (a && !a.tagName) try {

for (c in a) a[c] && ("object" == typeof a[c] && d.deleteObject(a[c], b - 1), "function" != typeof a[c] && (a[c] = null))

} catch (e) {

}

}

};

d.bounce = function (a, b, c, e, h) {

return (b /= h) < 1 / 2.75 ? 7.5625 \* e \* b \* b + c : b < 2 / 2.75 ? e \* (7.5625 \* (b -= 1.5 / 2.75) \* b + .75) + c : b < 2.5 / 2.75 ? e \* (7.5625 \* (b -= 2.25 / 2.75) \* b + .9375) + c : e \* (7.5625 \* (b -= 2.625 / 2.75) \* b + .984375) + c

};

d.easeInSine = function (a, b, c, e, h) {

return -e \*

Math.cos(b / h \* (Math.PI / 2)) + e + c

};

d.easeOutSine = function (a, b, c, e, h) {

return e \* Math.sin(b / h \* (Math.PI / 2)) + c

};

d.easeOutElastic = function (a, b, c, e, h) {

a = 1.70158;

var d = 0, g = e;

if (0 === b) return c;

if (1 == (b /= h)) return c + e;

d || (d = .3 \* h);

g < Math.abs(e) ? (g = e, a = d / 4) : a = d / (2 \* Math.PI) \* Math.asin(e / g);

return g \* Math.pow(2, -10 \* b) \* Math.sin(2 \* (b \* h - a) \* Math.PI / d) + e + c

};

d.fixStepE = function (a) {

a = a.toExponential(0).split("e");

var b = Number(a[1]);

9 == Number(a[0]) && b++;

return d.generateNumber(1, b)

};

d.generateNumber = function (a, b) {

var c = "", e;

e = 0 > b ? Math.abs(b) - 1 : Math.abs(b);

var h;

for (h = 0; h < e; h++) c += "0";

return 0 > b ? Number("0." + c + String(a)) : Number(String(a) + c)

};

d.setCN = function (a, b, c, e) {

if (a.addClassNames && b && (b = b.node) && c) {

var h = b.getAttribute("class");

a = a.classNamePrefix + "-";

e && (a = "");

h ? b.setAttribute("class", h + " " + a + c) : b.setAttribute("class", a + c)

}

};

d.parseDefs = function (a, b) {

for (var c in a) {

var e = typeof a[c];

if (0 < a[c].length && "object" == e) for (var h = 0; h < a[c].length; h++) e = document.createElementNS(d.SVG\_NS, c), b.appendChild(e), d.parseDefs(a[c][h],

e); else "object" == e ? (e = document.createElementNS(d.SVG\_NS, c), b.appendChild(e), d.parseDefs(a[c], e)) : b.setAttribute(c, a[c])

}

}

})();

(function () {

var d = window.AmCharts;

d.AxisBase = d.Class({

construct: function (a) {

this.createEvents("clickItem", "rollOverItem", "rollOutItem");

this.viY = this.viX = this.titleDY = this.y = this.x = this.dy = this.dx = 0;

this.axisThickness = 1;

this.axisColor = "#000000";

this.axisAlpha = 1;

this.gridCount = this.tickLength = 5;

this.gridAlpha = .15;

this.gridThickness = 1;

this.gridColor = "#000000";

this.dashLength = 0;

this.labelFrequency = 1;

this.showLastLabel = this.showFirstLabel = !0;

this.fillColor = "#FFFFFF";

this.fillAlpha = 0;

this.labelsEnabled =

!0;

this.labelRotation = 0;

this.autoGridCount = !0;

this.offset = 0;

this.guides = [];

this.visible = !0;

this.counter = 0;

this.guides = [];

this.ignoreAxisWidth = this.inside = !1;

this.minHorizontalGap = 75;

this.minVerticalGap = 35;

this.titleBold = !0;

this.minorGridEnabled = !1;

this.minorGridAlpha = .07;

this.autoWrap = !1;

this.titleAlign = "middle";

this.labelOffset = 0;

this.bcn = "axis-";

this.centerLabels = !1;

this.periods = [{period: "ss", count: 1}, {period: "ss", count: 5}, {

period: "ss",

count: 10

}, {period: "ss", count: 30}, {period: "mm", count: 1}, {

period: "mm",

count: 5

}, {period: "mm", count: 10}, {period: "mm", count: 30}, {period: "hh", count: 1}, {

period: "hh",

count: 3

}, {period: "hh", count: 6}, {period: "hh", count: 12}, {period: "DD", count: 1}, {

period: "DD",

count: 2

}, {period: "DD", count: 3}, {period: "DD", count: 4}, {period: "DD", count: 5}, {

period: "WW",

count: 1

}, {period: "MM", count: 1}, {period: "MM", count: 2}, {period: "MM", count: 3}, {

period: "MM",

count: 6

}, {period: "YYYY", count: 1}, {period: "YYYY", count: 2}, {period: "YYYY", count: 5}, {

period: "YYYY",

count: 10

}, {period: "YYYY", count: 50}, {

period: "YYYY",

count: 100

}];

this.dateFormats = [{period: "fff", format: "JJ:NN:SS"}, {period: "ss", format: "JJ:NN:SS"}, {

period: "mm",

format: "JJ:NN"

}, {period: "hh", format: "JJ:NN"}, {period: "DD", format: "MMM DD"}, {

period: "WW",

format: "MMM DD"

}, {period: "MM", format: "MMM"}, {period: "YYYY", format: "YYYY"}];

this.nextPeriod = {fff: "ss", ss: "mm", mm: "hh", hh: "DD", DD: "MM", MM: "YYYY"};

d.applyTheme(this, a, "AxisBase")

}, zoom: function (a, b) {

this.start = a;

this.end = b;

this.dataChanged = !0;

this.draw()

}, fixAxisPosition: function () {

var a = this.position;

"H" ==

this.orientation ? ("left" == a && (a = "bottom"), "right" == a && (a = "top")) : ("bottom" == a && (a = "left"), "top" == a && (a = "right"));

this.position = a

}, draw: function () {

var a = this.chart;

this.allLabels = [];

this.counter = 0;

this.destroy();

this.fixAxisPosition();

this.labels = [];

var b = a.container, c = b.set();

a.gridSet.push(c);

this.set = c;

b = b.set();

a.axesLabelsSet.push(b);

this.labelsSet = b;

this.axisLine = new this.axisRenderer(this);

this.autoGridCount ? ("V" == this.orientation ? (a = this.height / this.minVerticalGap, 3 > a && (a = 3)) : a = this.width /

this.minHorizontalGap, this.gridCountR = Math.max(a, 1)) : this.gridCountR = this.gridCount;

this.axisWidth = this.axisLine.axisWidth;

this.addTitle()

}, setOrientation: function (a) {

this.orientation = a ? "H" : "V"

}, addTitle: function () {

var a = this.title;

this.titleLabel = null;

if (a) {

var b = this.chart, c = this.titleColor;

void 0 === c && (c = b.color);

var e = this.titleFontSize;

isNaN(e) && (e = b.fontSize + 1);

a = d.text(b.container, a, c, b.fontFamily, e, this.titleAlign, this.titleBold);

d.setCN(b, a, this.bcn + "title");

this.titleLabel = a

}

}, positionTitle: function () {

var a =

this.titleLabel;

if (a) {

var b, c, e = this.labelsSet, h = {};

0 < e.length() ? h = e.getBBox() : (h.x = 0, h.y = 0, h.width = this.viW, h.height = this.viH, d.VML && (h.y += this.y, h.x += this.x));

e.push(a);

var e = h.x, f = h.y;

d.VML && (this.rotate ? e -= this.x : f -= this.y);

var g = h.width, h = h.height, k = this.viW, l = this.viH, m = 0, n = a.getBBox().height / 2,

p = this.inside, q = this.titleAlign;

switch (this.position) {

case "top":

b = "left" == q ? -1 : "right" == q ? k : k / 2;

c = f - 10 - n;

break;

case "bottom":

b = "left" == q ? -1 : "right" == q ? k : k / 2;

c = f + h + 10 + n;

break;

case "left":

b = e - 10 - n;

p &&

(b -= 5);

m = -90;

c = ("left" == q ? l + 1 : "right" == q ? -1 : l / 2) + this.titleDY;

break;

case "right":

b = e + g + 10 + n, p && (b += 7), c = ("left" == q ? l + 2 : "right" == q ? -2 : l / 2) + this.titleDY, m = -90

}

this.marginsChanged ? (a.translate(b, c), this.tx = b, this.ty = c) : a.translate(this.tx, this.ty);

this.marginsChanged = !1;

isNaN(this.titleRotation) || (m = this.titleRotation);

0 !== m && a.rotate(m)

}

}, pushAxisItem: function (a, b) {

var c = this, e = a.graphics();

0 < e.length() && (b ? c.labelsSet.push(e) : c.set.push(e));

if (e = a.getLabel()) this.labelsSet.push(e), e.click(function (b) {

c.handleMouse(b,

a, "clickItem")

}).mouseover(function (b) {

c.handleMouse(b, a, "rollOverItem")

}).mouseout(function (b) {

c.handleMouse(b, a, "rollOutItem")

})

}, handleMouse: function (a, b, c) {

this.fire(c, {

type: c,

value: b.value,

serialDataItem: b.serialDataItem,

axis: this,

target: b.label,

chart: this.chart,

event: a

})

}, addGuide: function (a) {

for (var b = this.guides, c = !1, e = b.length, h = 0; h < b.length; h++) b[h] == a && (c = !0, e = h);

a = d.processObject(a, d.Guide, this.theme);

a.id || (a.id = "guideAuto" + e + "\_" + (new Date).getTime());

c || b.push(a)

}, removeGuide: function (a) {

var b =

this.guides, c;

for (c = 0; c < b.length; c++) b[c] == a && b.splice(c, 1)

}, handleGuideOver: function (a) {

clearTimeout(this.chart.hoverInt);

var b = a.graphics.getBBox(), c = b.x + b.width / 2, b = b.y + b.height / 2, e = a.fillColor;

void 0 === e && (e = a.lineColor);

this.chart.showBalloon(a.balloonText, e, !0, c, b)

}, handleGuideOut: function () {

this.chart.hideBalloon()

}, addEventListeners: function (a, b) {

var c = this;

a.mouseover(function () {

c.handleGuideOver(b)

});

a.touchstart(function () {

c.handleGuideOver(b)

});

a.mouseout(function () {

c.handleGuideOut(b)

})

},

getBBox: function () {

var a = this.labelsSet.getBBox();

d.VML || (a = {x: a.x + this.x, y: a.y + this.y, width: a.width, height: a.height});

return a

}, destroy: function () {

d.remove(this.set);

d.remove(this.labelsSet);

var a = this.axisLine;

a && d.remove(a.set);

d.remove(this.grid0)

}, chooseMinorFrequency: function (a) {

for (var b = 10; 0 < b; b--) if (a / b == Math.round(a / b)) return a / b

}, parseDatesDraw: function () {

var a, b = this.chart, c = this.showFirstLabel, e = this.showLastLabel, h, f = "",

g = d.extractPeriod(this.minPeriod), k = d.getPeriodDuration(g.period,

g.count), l, m, n, p, q, r = this.firstDayOfWeek, u = this.boldPeriodBeginning;

a = this.minorGridEnabled;

var t, z = this.gridAlpha, y, v = this.choosePeriod(0), w = v.period, v = v.count,

A = d.getPeriodDuration(w, v);

A < k && (w = g.period, v = g.count, A = k);

g = w;

"WW" == g && (g = "DD");

this.stepWidth = this.getStepWidth(this.timeDifference);

var B = Math.ceil(this.timeDifference / A) + 5,

C = l = d.resetDateToMin(new Date(this.startTime - A), w, v, r).getTime();

if (g == w && 1 == v && this.centerLabelOnFullPeriod || this.autoWrap || this.centerLabels) n = A \* this.stepWidth, this.autoWrap &&

!this.centerLabels && (n = -n);

this.cellWidth = k \* this.stepWidth;

p = Math.round(l / A);

k = -1;

p / 2 == Math.round(p / 2) && (k = -2, l -= A);

p = this.firstTime;

var D = 0, J = 0;

a && 1 < v && (t = this.chooseMinorFrequency(v), y = d.getPeriodDuration(w, t));

if (0 < this.gridCountR) for (B - 5 - k > this.autoRotateCount && !isNaN(this.autoRotateAngle) && (this.labelRotationR = this.autoRotateAngle), a = k; a <= B; a++) {

q = p + A \* (a + Math.floor((C - p) / A)) - D;

"DD" == w && (q += 36E5);

q = d.resetDateToMin(new Date(q), w, v, r).getTime();

"MM" == w && (h = (q - l) / A, 1.5 <= (q - l) / A && (q = q - (h - 1) \* A + d.getPeriodDuration("DD",

3), q = d.resetDateToMin(new Date(q), w, 1).getTime(), D += A));

h = (q - this.startTime) \* this.stepWidth;

if ("radar" == b.type) {

if (0 > h || h > this.axisWidth) continue;

h = this.y + (this.axisWidth - h)

} else this.rotate ? (h += this.x - this.viX, "date" == this.type && "middle" == this.gridPosition && (J = -A \* this.stepWidth / 2)) : h = "date" == this.type ? this.axisWidth - h + (this.y - this.viY) : h + (this.y - this.viY);

f = !1;

this.nextPeriod[g] && (f = this.checkPeriodChange(this.nextPeriod[g], 1, q, l, g));

l = !1;

f && this.markPeriodChange ? (f = this.dateFormatsObject[this.nextPeriod[g]],

this.twoLineMode && (f = this.dateFormatsObject[g] + "\n" + f, f = d.fixBrakes(f)), l = !0) : f = this.dateFormatsObject[g];

u || (l = !1);

this.currentDateFormat = f;

f = d.formatDate(new Date(q), f, b);

if (a == k && !c || a == B && !e) f = " ";

this.labelFunction && (f = this.labelFunction(f, new Date(q), this, w, v, m).toString());

this.boldLabels && (l = !0);

m = new this.axisItemRenderer(this, h, f, !1, n, J, !1, l);

this.pushAxisItem(m);

m = l = q;

if (!isNaN(t)) for (h = 1; h < v; h += t) this.gridAlpha = this.minorGridAlpha, f = q + y \* h, f = d.resetDateToMin(new Date(f), w, t, r).getTime(),

f = new this.axisItemRenderer(this, (f - this.startTime) \* this.stepWidth, void 0, void 0, void 0, void 0, void 0, void 0, void 0, !0), this.pushAxisItem(f);

this.gridAlpha = z

}

}, choosePeriod: function (a) {

var b = d.getPeriodDuration(this.periods[a].period, this.periods[a].count),

c = Math.ceil(this.timeDifference / b), e = this.periods;

return this.timeDifference < b && 0 < a ? e[a - 1] : c <= this.gridCountR ? e[a] : a + 1 < e.length ? this.choosePeriod(a + 1) : e[a]

}, getStepWidth: function (a) {

var b;

this.startOnAxis ? (b = this.axisWidth / (a - 1), 1 == a && (b = this.axisWidth)) :

b = this.axisWidth / a;

return b

}, timeZoom: function (a, b) {

this.startTime = a;

this.endTime = b

}, minDuration: function () {

var a = d.extractPeriod(this.minPeriod);

return d.getPeriodDuration(a.period, a.count)

}, checkPeriodChange: function (a, b, c, e, h) {

c = new Date(c);

var f = new Date(e), g = this.firstDayOfWeek;

e = b;

"DD" == a && (b = 1);

c = d.resetDateToMin(c, a, b, g).getTime();

b = d.resetDateToMin(f, a, b, g).getTime();

return "DD" == a && "hh" != h && c - b < d.getPeriodDuration(a, e) ? !1 : c != b ? !0 : !1

}, generateDFObject: function () {

this.dateFormatsObject = {};

var a;

for (a = 0; a < this.dateFormats.length; a++) {

var b = this.dateFormats[a];

this.dateFormatsObject[b.period] = b.format

}

}

})

})();

(function () {

var d = window.AmCharts;

d.ValueAxis = d.Class({

inherits: d.AxisBase, construct: function (a) {

this.cname = "ValueAxis";

this.createEvents("axisChanged", "logarithmicAxisFailed", "axisSelfZoomed", "axisZoomed");

d.ValueAxis.base.construct.call(this, a);

this.dataChanged = !0;

this.stackType = "none";

this.position = "left";

this.unitPosition = "right";

this.includeAllValues = this.recalculateToPercents = this.includeHidden = this.includeGuidesInMinMax = this.integersOnly = !1;

this.durationUnits = {DD: "d. ", hh: ":", mm: ":", ss: ""};

this.scrollbar = !1;

this.baseValue = 0;

this.radarCategoriesEnabled = !0;

this.gridType = "polygons";

this.useScientificNotation = !1;

this.axisTitleOffset = 10;

this.pointPosition = "axis";

this.minMaxMultiplier = 1;

this.logGridLimit = 2;

this.totalTextOffset = this.treatZeroAs = 0;

this.minPeriod = "ss";

d.applyTheme(this, a, this.cname)

}, updateData: function () {

0 >= this.gridCountR && (this.gridCountR = 1);

this.totals = [];

this.data = this.chart.chartData;

var a = this.chart;

"xy" != a.type && (this.stackGraphs("smoothedLine"), this.stackGraphs("line"),

this.stackGraphs("column"), this.stackGraphs("step"));

this.recalculateToPercents && this.recalculate();

this.synchronizationMultiplier && this.synchronizeWith ? (d.isString(this.synchronizeWith) && (this.synchronizeWith = a.getValueAxisById(this.synchronizeWith)), this.synchronizeWith && (this.synchronizeWithAxis(this.synchronizeWith), this.foundGraphs = !0)) : (this.foundGraphs = !1, this.getMinMax())

}, draw: function () {

d.ValueAxis.base.draw.call(this);

var a = this.chart, b = this.set;

this.labelRotationR = this.labelRotation;

d.setCN(a,

this.set, "value-axis value-axis-" + this.id);

d.setCN(a, this.labelsSet, "value-axis value-axis-" + this.id);

d.setCN(a, this.axisLine.axisSet, "value-axis value-axis-" + this.id);

var c = this.type;

"duration" == c && (this.duration = "ss");

!0 === this.dataChanged && (this.updateData(), this.dataChanged = !1);

"date" == c && (this.logarithmic = !1, this.min = this.minRR, this.max = this.maxRR, this.getDateMinMax());

if (this.logarithmic) {

var e = this.treatZeroAs, h = this.getMin(0, this.data.length - 1);

this.minReal < h && (this.minReal = h);

isNaN(this.minReal) &&

(this.minReal = h);

0 < e && 0 === h && (this.minReal = h = e);

if (0 >= h || 0 >= this.minimum) {

this.fire("logarithmicAxisFailed", {type: "logarithmicAxisFailed", chart: a});

return

}

}

this.grid0 = null;

var f, g, k = a.dx, l = a.dy, e = !1, h = this.logarithmic;

if (isNaN(this.min) || isNaN(this.max) || !this.foundGraphs || Infinity == this.min || -Infinity == this.max) e = !0; else {

var m = this.labelFrequency, n = this.showFirstLabel, p = this.showLastLabel, q = 1;

f = 0;

this.minCalc = this.min;

this.maxCalc = this.max;

this.strictMinMax && (isNaN(this.minimum) || (this.min = this.minimum),

isNaN(this.maximum) || (this.max = this.maximum));

isNaN(this.minZoom) || (this.min = this.minZoom);

isNaN(this.maxZoom) || (this.max = this.maxZoom);

var r = Math.round((this.maxCalc - this.minCalc) / this.step) + 1, u;

!0 === h ? (u = Math.log(this.max) \* Math.LOG10E - Math.log(this.minReal) \* Math.LOG10E, this.stepWidth = this.axisWidth / u, u > this.logGridLimit && (r = Math.ceil(Math.log(this.max) \* Math.LOG10E) + 1, f = Math.round(Math.log(this.minReal) \* Math.LOG10E), r > this.gridCountR && (q = Math.ceil(r / this.gridCountR)))) : this.stepWidth = this.axisWidth /

(this.max - this.min);

var t = 0;

1 > this.step && -1 < this.step && (t = d.getDecimals(this.step));

this.integersOnly && (t = 0);

t > this.maxDecCount && (t = this.maxDecCount);

var z = this.precision;

isNaN(z) || (t = z);

this.max = d.roundTo(this.max, this.maxDecCount);

this.min = d.roundTo(this.min, this.maxDecCount);

g = {};

g.precision = t;

g.decimalSeparator = a.nf.decimalSeparator;

g.thousandsSeparator = a.nf.thousandsSeparator;

this.numberFormatter = g;

var y, v = this.guides, w = v.length;

if (0 < w) {

var A = this.fillAlpha;

for (g = this.fillAlpha = 0; g < w; g++) {

var B =

v[g], C = NaN, D = B.above;

isNaN(B.toValue) || (C = this.getCoordinate(B.toValue), y = new this.axisItemRenderer(this, C, "", !0, NaN, NaN, B), this.pushAxisItem(y, D));

var J = NaN;

isNaN(B.value) || (J = this.getCoordinate(B.value), y = new this.axisItemRenderer(this, J, B.label, !0, NaN, (C - J) / 2, B), this.pushAxisItem(y, D));

isNaN(C - J) || (y = new this.guideFillRenderer(this, J, C, B), this.pushAxisItem(y, D), y = y.graphics(), B.graphics = y, B.balloonText && this.addEventListeners(y, B))

}

this.fillAlpha = A

}

this.exponential = !1;

for (g = f; g < r; g += q) v = d.roundTo(this.step \*

g + this.min, t), -1 != String(v).indexOf("e") && (this.exponential = !0);

this.duration && (this.maxInterval = d.getMaxInterval(this.max, this.duration));

var t = this.step, E, v = this.minorGridAlpha;

this.minorGridEnabled && (E = this.getMinorGridStep(t, this.stepWidth \* t));

if ("date" == c) this.generateDFObject(), this.timeDifference = this.max - this.min, this.maxTime = this.lastTime = this.max, this.startTime = this.firstTime = this.min, this.parseDatesDraw(); else for (r >= this.autoRotateCount && !isNaN(this.autoRotateAngle) && (this.labelRotationR =

this.autoRotateAngle), g = f; g < r; g += q) if (c = t \* g + this.minCalc, h && this.max - this.min > 10 \* this.min && (c -= this.min), c = d.roundTo(c, this.maxDecCount + 1), !this.integersOnly || Math.round(c) == c) if (isNaN(z) || Number(d.toFixed(c, z)) == c) {

!0 === h && (0 === c && (c = this.minReal), u > this.logGridLimit && (c = Math.pow(10, g)));

y = this.formatValue(c, !1, g);

Math.round(g / m) != g / m && (y = void 0);

if (0 === g && !n || g == r - 1 && !p) y = " ";

f = this.getCoordinate(c);

var O;

this.rotate && this.autoWrap && (O = this.stepWidth \* t - 10);

y = new this.axisItemRenderer(this, f, y, void 0,

O, void 0, void 0, this.boldLabels);

this.pushAxisItem(y);

if (c == this.baseValue && "radar" != a.type) {

var I, W, A = this.viW, B = this.viH;

y = this.viX;

w = this.viY;

"H" == this.orientation ? 0 <= f && f <= A + 1 && (I = [f, f, f + k], W = [B, 0, l]) : 0 <= f && f <= B + 1 && (I = [0, A, A + k], W = [f, f, f + l]);

I && (f = d.fitToBounds(2 \* this.gridAlpha, 0, 1), f = d.line(a.container, I, W, this.gridColor, f, 1, this.dashLength), f.translate(y, w), this.grid0 = f, a.axesSet.push(f), f.toBack(), d.setCN(a, f, this.bcn + "zero-grid-" + this.id), d.setCN(a, f, this.bcn + "zero-grid"))

}

if (!isNaN(E) && 0 <

v && g < r - 1) {

f = this.gridAlpha;

this.gridAlpha = this.minorGridAlpha;

for (y = 1; y < t / E; y++) w = this.getCoordinate(c + E \* y), w = new this.axisItemRenderer(this, w, "", !1, 0, 0, !1, !1, 0, !0), this.pushAxisItem(w);

this.gridAlpha = f

}

}

u = this.baseValue;

this.min > this.baseValue && this.max > this.baseValue && (u = this.min);

this.min < this.baseValue && this.max < this.baseValue && (u = this.max);

h && u < this.minReal && (u = this.minReal);

this.baseCoord = this.getCoordinate(u);

u = {type: "axisChanged", target: this, chart: a};

u.min = h ? this.minReal : this.min;

u.max = this.max;

this.fire("axisChanged", u);

this.axisCreated = !0

}

h = this.axisLine.set;

u = this.labelsSet;

this.positionTitle();

"radar" != a.type ? (a = this.viX, E = this.viY, b.translate(a, E), u.translate(a, E)) : h.toFront();

!this.visible || e ? (b.hide(), h.hide(), u.hide()) : (b.show(), h.show(), u.show());

this.axisY = this.y - this.viY;

this.axisX = this.x - this.viX

}, getDateMinMax: function () {

this.minimumDate && (this.minimumDate instanceof Date || (this.minimumDate = d.getDate(this.minimumDate, this.chart.dataDateFormat, "fff")), this.min = this.minimumDate.getTime());

this.maximumDate && (this.maximumDate instanceof Date || (this.maximumDate = d.getDate(this.maximumDate, this.chart.dataDateFormat, "fff")), this.max = this.maximumDate.getTime())

}, formatValue: function (a, b, c) {

var e = this.exponential, h = this.logarithmic, f = this.numberFormatter, g = this.chart;

!0 === this.logarithmic && (e = -1 != String(a).indexOf("e") ? !0 : !1);

this.useScientificNotation && (e = !0);

this.usePrefixes && (e = !1);

e ? (c = -1 == String(a).indexOf("e") ? a.toExponential(15) : String(a), e = c.split("e"), c = Number(e[0]), e = Number(e[1]),

c = d.roundTo(c, 14), 10 == c && (c = 1, e += 1), c = c + "e" + e, 0 === a && (c = "0"), 1 == a && (c = "1")) : (h && (e = String(a).split("."), e[1] ? (f.precision = e[1].length, 0 > c && (f.precision = Math.abs(c)), b && 1 < a && (f.precision = 0)) : f.precision = -1), c = this.usePrefixes ? d.addPrefix(a, g.prefixesOfBigNumbers, g.prefixesOfSmallNumbers, f, !b) : d.formatNumber(a, f, f.precision));

this.duration && (b && (f.precision = 0), c = d.formatDuration(a, this.duration, "", this.durationUnits, this.maxInterval, f));

"date" == this.type && (c = d.formatDate(new Date(a), this.currentDateFormat,

g));

this.recalculateToPercents ? c += "%" : (b = this.unit) && (c = "left" == this.unitPosition ? b + c : c + b);

this.labelFunction && (c = this.labelFunction(a, c, this).toString());

return c

}, getMinorGridStep: function (a, b) {

var c = [5, 4, 2];

60 > b && c.shift();

for (var e = Math.floor(Math.log(Math.abs(a)) \* Math.LOG10E), h = 0; h < c.length; h++) {

var d = a / c[h], g = Math.floor(Math.log(Math.abs(d)) \* Math.LOG10E);

if (!(1 < Math.abs(e - g))) if (1 > a) {

if (g = Math.pow(10, -g) \* d, g == Math.round(g)) return d

} else if (d == Math.round(d)) return d

}

}, stackGraphs: function (a) {

var b =

this.stackType;

"stacked" == b && (b = "regular");

"line" == b && (b = "none");

"100% stacked" == b && (b = "100%");

this.stackType = b;

var c = [], e = [], h = [], f = [], g, k = this.chart.graphs, l, m, n, p, q = this.baseValue, r = !1;

if ("line" == a || "step" == a || "smoothedLine" == a) r = !0;

if (r && ("regular" == b || "100%" == b)) for (p = 0; p < k.length; p++) n = k[p], n.hidden || (m = n.type, n.chart == this.chart && n.valueAxis == this && a == m && n.stackable && (l && (n.stackGraph = l), l = n));

for (l = this.start; l <= this.end; l++) {

var u = 0;

for (p = 0; p < k.length; p++) if (n = k[p], n.hidden) n.newStack && (h[l] =

NaN, e[l] = NaN); else if (m = n.type, n.chart == this.chart && n.valueAxis == this && a == m && n.stackable) if (m = this.data[l].axes[this.id].graphs[n.id], g = m.values.value, isNaN(g)) n.newStack && (h[l] = NaN, e[l] = NaN); else {

var t = d.getDecimals(g);

u < t && (u = t);

isNaN(f[l]) ? f[l] = Math.abs(g) : f[l] += Math.abs(g);

f[l] = d.roundTo(f[l], u);

t = n.fillToGraph;

r && t && (t = this.data[l].axes[this.id].graphs[t.id]) && (m.values.open = t.values.value);

"regular" == b && (r && (isNaN(c[l]) ? (c[l] = g, m.values.close = g, m.values.open = this.baseValue) : (isNaN(g) ? m.values.close =

c[l] : m.values.close = g + c[l], m.values.open = c[l], c[l] = m.values.close)), "column" == a && (n.newStack && (h[l] = NaN, e[l] = NaN), m.values.close = g, 0 > g ? (m.values.close = g, isNaN(e[l]) ? m.values.open = q : (m.values.close += e[l], m.values.open = e[l]), e[l] = m.values.close) : (m.values.close = g, isNaN(h[l]) ? m.values.open = q : (m.values.close += h[l], m.values.open = h[l]), h[l] = m.values.close)))

}

}

for (l = this.start; l <= this.end; l++) for (p = 0; p < k.length; p++) (n = k[p], n.hidden) ? n.newStack && (h[l] = NaN, e[l] = NaN) : (m = n.type, n.chart == this.chart && n.valueAxis ==

this && a == m && n.stackable && (m = this.data[l].axes[this.id].graphs[n.id], g = m.values.value, isNaN(g) || (c = g / f[l] \* 100, m.values.percents = c, m.values.total = f[l], n.newStack && (h[l] = NaN, e[l] = NaN), "100%" == b && (isNaN(e[l]) && (e[l] = 0), isNaN(h[l]) && (h[l] = 0), 0 > c ? (m.values.close = d.fitToBounds(c + e[l], -100, 100), m.values.open = e[l], e[l] = m.values.close) : (m.values.close = d.fitToBounds(c + h[l], -100, 100), m.values.open = h[l], h[l] = m.values.close)))))

}, recalculate: function () {

var a = this.chart, b = a.graphs, c;

for (c = 0; c < b.length; c++) {

var e =

b[c];

if (e.valueAxis == this) {

var h = "value";

if ("candlestick" == e.type || "ohlc" == e.type) h = "open";

var f, g, k = this.end + 2, k = d.fitToBounds(this.end + 1, 0, this.data.length - 1),

l = this.start;

0 < l && l--;

var m;

g = this.start;

e.compareFromStart && (g = 0);

if (!isNaN(a.startTime) && (m = a.categoryAxis)) {

var n = m.minDuration(), n = new Date(a.startTime + n / 2),

p = d.resetDateToMin(new Date(a.startTime), m.minPeriod).getTime();

d.resetDateToMin(new Date(n), m.minPeriod).getTime() > p && g++

}

if (m = a.recalculateFromDate) m = d.getDate(m, a.dataDateFormat,

"fff"), g = a.getClosestIndex(a.chartData, "time", m.getTime(), !0, 0, a.chartData.length), k = a.chartData.length - 1;

for (m = g; m <= k && (g = this.data[m].axes[this.id].graphs[e.id], f = g.values[h], isNaN(f)); m++) ;

this.recBaseValue = f;

for (h = l; h <= k; h++) {

g = this.data[h].axes[this.id].graphs[e.id];

g.percents = {};

var l = g.values, q;

for (q in l) g.percents[q] = "percents" != q ? l[q] / f \* 100 - 100 : l[q]

}

}

}

}, getMinMax: function () {

var a = !1, b = this.chart, c = b.graphs, e;

for (e = 0; e < c.length; e++) {

var h = c[e].type;

("line" == h || "step" == h || "smoothedLine" ==

h) && this.expandMinMax && (a = !0)

}

a && (0 < this.start && this.start--, this.end < this.data.length - 1 && this.end++);

"serial" == b.type && (!0 !== b.categoryAxis.parseDates || a || this.end < this.data.length - 1 && this.end++);

this.includeAllValues && (this.start = 0, this.end = this.data.length - 1);

a = this.minMaxMultiplier;

this.min = this.getMin(this.start, this.end);

this.max = this.getMax();

this.minRR = this.min;

this.maxRR = this.max;

a = (this.max - this.min) \* (a - 1);

this.min -= a;

this.max += a;

a = this.guides.length;

if (this.includeGuidesInMinMax && 0 < a) for (b =

0; b < a; b++) c = this.guides[b], c.toValue < this.min && (this.min = c.toValue), c.value < this.min && (this.min = c.value), c.toValue > this.max && (this.max = c.toValue), c.value > this.max && (this.max = c.value);

isNaN(this.minimum) || (this.min = this.minimum);

isNaN(this.maximum) || (this.max = this.maximum);

"date" == this.type && this.getDateMinMax();

this.min > this.max && (a = this.max, this.max = this.min, this.min = a);

isNaN(this.minTemp) || (this.min = this.minTemp);

isNaN(this.maxTemp) || (this.max = this.maxTemp);

this.minReal = this.min;

this.maxReal = this.max;

0 === this.min && 0 === this.max && (this.max = 9);

this.min > this.max && (this.min = this.max - 1);

a = this.min;

b = this.max;

c = this.max - this.min;

e = 0 === c ? Math.pow(10, Math.floor(Math.log(Math.abs(this.max)) \* Math.LOG10E)) / 10 : Math.pow(10, Math.floor(Math.log(Math.abs(c)) \* Math.LOG10E)) / 10;

isNaN(this.maximum) && isNaN(this.maxTemp) && (this.max = Math.ceil(this.max / e) \* e + e);

isNaN(this.minimum) && isNaN(this.minTemp) && (this.min = Math.floor(this.min / e) \* e - e);

0 > this.min && 0 <= a && (this.min = 0);

0 < this.max && 0 >= b && (this.max = 0);

"100%" == this.stackType &&

(this.min = 0 > this.min ? -100 : 0, this.max = 0 > this.max ? 0 : 100);

c = this.max - this.min;

e = Math.pow(10, Math.floor(Math.log(Math.abs(c)) \* Math.LOG10E)) / 10;

this.step = Math.ceil(c / this.gridCountR / e) \* e;

c = Math.pow(10, Math.floor(Math.log(Math.abs(this.step)) \* Math.LOG10E));

c = d.fixStepE(c);

e = Math.ceil(this.step / c);

5 < e && (e = 10);

5 >= e && 2 < e && (e = 5);

this.step = Math.ceil(this.step / (c \* e)) \* c \* e;

1 > c ? (this.maxDecCount = Math.abs(Math.log(Math.abs(c)) \* Math.LOG10E), this.maxDecCount = Math.round(this.maxDecCount), this.step = d.roundTo(this.step,

this.maxDecCount + 1)) : this.maxDecCount = 0;

this.min = this.step \* Math.floor(this.min / this.step);

this.max = this.step \* Math.ceil(this.max / this.step);

0 > this.min && 0 <= a && (this.min = 0);

0 < this.max && 0 >= b && (this.max = 0);

1 < this.minReal && 1 < this.max - this.minReal && (this.minReal = Math.floor(this.minReal));

c = Math.pow(10, Math.floor(Math.log(Math.abs(this.minReal)) \* Math.LOG10E));

0 === this.min && (this.minReal = c);

0 === this.min && 1 < this.minReal && (this.minReal = 1);

0 < this.min && 0 < this.minReal - this.step && (this.minReal = this.min + this.step <

this.minReal ? this.min + this.step : this.min);

this.logarithmic && (2 < Math.log(b) \* Math.LOG10E - Math.log(a) \* Math.LOG10E ? (this.minReal = this.min = Math.pow(10, Math.floor(Math.log(Math.abs(a)) \* Math.LOG10E)), this.max = Math.pow(10, Math.ceil(Math.log(Math.abs(b)) \* Math.LOG10E))) : (b = Math.pow(10, Math.floor(Math.log(Math.abs(this.min)) \* Math.LOG10E)) / 10, a = Math.pow(10, Math.floor(Math.log(Math.abs(a)) \* Math.LOG10E)) / 10, b < a && (this.minReal = this.min = 10 \* a)))

}, getMin: function (a, b) {

var c, e;

for (e = a; e <= b; e++) {

var d = this.data[e].axes[this.id].graphs,

f;

for (f in d) if (d.hasOwnProperty(f)) {

var g = this.chart.getGraphById(f);

if (g.includeInMinMax && (!g.hidden || this.includeHidden)) {

isNaN(c) && (c = Infinity);

this.foundGraphs = !0;

g = d[f].values;

this.recalculateToPercents && (g = d[f].percents);

var k;

if (this.minMaxField) k = g[this.minMaxField], k < c && (c = k); else for (var l in g) g.hasOwnProperty(l) && "percents" != l && "total" != l && (k = g[l], k < c && (c = k))

}

}

}

return c

}, getMax: function () {

var a, b;

for (b = this.start; b <= this.end; b++) {

var c = this.data[b].axes[this.id].graphs, e;

for (e in c) if (c.hasOwnProperty(e)) {

var d =

this.chart.getGraphById(e);

if (d.includeInMinMax && (!d.hidden || this.includeHidden)) {

isNaN(a) && (a = -Infinity);

this.foundGraphs = !0;

d = c[e].values;

this.recalculateToPercents && (d = c[e].percents);

var f;

if (this.minMaxField) f = d[this.minMaxField], f > a && (a = f); else for (var g in d) d.hasOwnProperty(g) && "percents" != g && "total" != g && (f = d[g], f > a && (a = f))

}

}

}

return a

}, dispatchZoomEvent: function (a, b) {

var c = {type: "axisZoomed", startValue: a, endValue: b, target: this, chart: this.chart};

this.fire(c.type, c)

}, zoomOut: function () {

var a =

this.chart;

"xy" != a.type && (this.maxZoom = this.minZoom = void 0, a.updateAfterValueZoom())

}, zoomToValues: function (a, b) {

var c = this.chart;

"xy" == c.type ? (b < a && (c = b, b = a, a = c), a < this.min && (a = this.min), b > this.max && (b = this.max), c = {type: "axisSelfZoomed"}, c.chart = this.chart, c.valueAxis = this, c.multiplier = this.axisWidth / Math.abs(this.getCoordinate(b) - this.getCoordinate(a)), c.startValue = a, c.endValue = b, c.position = "V" == this.orientation ? this.reversed ? this.getCoordinate(a) : this.getCoordinate(b) : this.reversed ? this.getCoordinate(b) :

this.getCoordinate(a), this.fire(c.type, c)) : (this.minZoom = a, this.maxZoom = b, c.updateAfterValueZoom())

}, coordinateToValue: function (a) {

if (isNaN(a)) return NaN;

var b = this.axisWidth, c = this.stepWidth, e = this.reversed, d = this.rotate, f = this.min,

g = this.minReal;

return !0 === this.logarithmic ? Math.pow(10, (d ? !0 === e ? (b - a) / c : a / c : !0 === e ? a / c : (b - a) / c) + Math.log(g) \* Math.LOG10E) : !0 === e ? d ? f - (a - b) / c : a / c + f : d ? a / c + f : f - (a - b) / c

}, getCoordinate: function (a) {

if (isNaN(a)) return NaN;

var b = this.rotate, c = this.reversed, e = this.axisWidth, d =

this.stepWidth, f = this.min, g = this.minReal;

!0 === this.logarithmic ? (0 === a && (a = this.treatZeroAs), a = Math.log(a) \* Math.LOG10E - Math.log(g) \* Math.LOG10E, b = b ? !0 === c ? e - d \* a : d \* a : !0 === c ? d \* a : e - d \* a) : b = !0 === c ? b ? e - d \* (a - f) : d \* (a - f) : b ? d \* (a - f) : e - d \* (a - f);

b = this.rotate ? b + (this.x - this.viX) : b + (this.y - this.viY);

1E7 < Math.abs(b) && (b = 1E7 \* (b / Math.abs(b)));

return Math.round(b)

}, synchronizeWithAxis: function (a) {

this.synchronizeWith = a;

this.listenTo(this.synchronizeWith, "axisChanged", this.handleSynchronization)

}, handleSynchronization: function () {

if (this.synchronizeWith) {

d.isString(this.synchronizeWith) &&

(this.synchronizeWith = this.chart.getValueAxisById(this.synchronizeWith));

var a = this.synchronizeWith, b = a.min, c = a.max, a = a.step, e = this.synchronizationMultiplier;

e && (this.min = b \* e, this.max = c \* e, this.step = a \* e, b = Math.pow(10, Math.floor(Math.log(Math.abs(this.step)) \* Math.LOG10E)), b = Math.abs(Math.log(Math.abs(b)) \* Math.LOG10E), this.maxDecCount = b = Math.round(b), this.draw())

}

}

})

})();

(function () {

var d = window.AmCharts;

d.RecAxis = d.Class({

construct: function (a) {

var b = a.chart, c = a.axisThickness, e = a.axisColor, h = a.axisAlpha, f = a.offset, g = a.dx, k = a.dy,

l = a.viX, m = a.viY, n = a.viH, p = a.viW, q = b.container;

"H" == a.orientation ? (e = d.line(q, [0, p], [0, 0], e, h, c), this.axisWidth = a.width, "bottom" == a.position ? (k = c / 2 + f + n + m - 1, c = l) : (k = -c / 2 - f + m + k, c = g + l)) : (this.axisWidth = a.height, "right" == a.position ? (e = d.line(q, [0, 0, -g], [0, n, n - k], e, h, c), k = m + k, c = c / 2 + f + g + p + l - 1) : (e = d.line(q, [0, 0], [0, n], e, h, c), k = m, c = -c / 2 - f + l));

e.translate(c,

k);

c = b.container.set();

c.push(e);

b.axesSet.push(c);

d.setCN(b, e, a.bcn + "line");

this.axisSet = c;

this.set = e

}

})

})();

(function () {

var d = window.AmCharts;

d.RecItem = d.Class({

construct: function (a, b, c, e, h, f, g, k, l, m, n, p) {

b = Math.round(b);

var q = a.chart;

this.value = c;

void 0 == c && (c = "");

l || (l = 0);

void 0 == e && (e = !0);

var r = q.fontFamily, u = a.fontSize;

void 0 == u && (u = q.fontSize);

var t = a.color;

void 0 == t && (t = q.color);

void 0 !== n && (t = n);

var z = a.chart.container, y = z.set();

this.set = y;

var v = a.axisThickness, w = a.axisColor, A = a.axisAlpha, B = a.tickLength, C = a.gridAlpha,

D = a.gridThickness, J = a.gridColor, E = a.dashLength, O = a.fillColor, I = a.fillAlpha,

W = a.labelsEnabled;

n = a.labelRotationR;

var ea = a.counter, P = a.inside, oa = a.labelOffset, qa = a.dx, ka = a.dy, Qa = a.orientation,

ca = a.position, aa = a.previousCoord, V = a.viH, sa = a.viW, ba = a.offset, da, ta;

g ? (void 0 !== g.id && (p = q.classNamePrefix + "-guide-" + g.id), W = !0, isNaN(g.tickLength) || (B = g.tickLength), void 0 != g.lineColor && (J = g.lineColor), void 0 != g.color && (t = g.color), isNaN(g.lineAlpha) || (C = g.lineAlpha), isNaN(g.dashLength) || (E = g.dashLength), isNaN(g.lineThickness) || (D = g.lineThickness), !0 === g.inside && (P = !0, 0 < ba && (ba = 0)), isNaN(g.labelRotation) ||

(n = g.labelRotation), isNaN(g.fontSize) || (u = g.fontSize), g.position && (ca = g.position), void 0 !== g.boldLabel && (k = g.boldLabel), isNaN(g.labelOffset) || (oa = g.labelOffset)) : "" === c && (B = 0);

m && !isNaN(a.minorTickLength) && (B = a.minorTickLength);

var fa = "start";

0 < h && (fa = "middle");

a.centerLabels && (fa = "middle");

var R = n \* Math.PI / 180, X, wa, H = 0, x = 0, la = 0, ga = X = 0, Ja = 0;

"V" == Qa && (n = 0);

var Z;

W && "" !== c && (Z = a.autoWrap && 0 === n ? d.wrappedText(z, c, t, r, u, fa, k, Math.abs(h), 0) : d.text(z, c, t, r, u, fa, k), fa = Z.getBBox(), ga = fa.width, Ja = fa.height);

if ("H" == Qa) {

if (0 <= b && b <= sa + 1 && (0 < B && 0 < A && b + l <= sa + 1 && (da = d.line(z, [b + l, b + l], [0, B], w, A, D), y.push(da)), 0 < C && (ta = d.line(z, [b, b + qa, b + qa], [V, V + ka, ka], J, C, D, E), y.push(ta))), x = 0, H = b, g && 90 == n && P && (H -= u), !1 === e ? (fa = "start", x = "bottom" == ca ? P ? x + B : x - B : P ? x - B : x + B, H += 3, 0 < h && (H += h / 2 - 3, fa = "middle"), 0 < n && (fa = "middle")) : fa = "middle", 1 == ea && 0 < I && !g && !m && aa < sa && (e = d.fitToBounds(b, 0, sa), aa = d.fitToBounds(aa, 0, sa), X = e - aa, 0 < X && (wa = d.rect(z, X, a.height, O, I), wa.translate(e - X + qa, ka), y.push(wa))), "bottom" == ca ? (x += V + u / 2 + ba, P ? (0 < n ?

(x = V - ga / 2 \* Math.sin(R) - B - 3, H += ga / 2 \* Math.cos(R) - 4 + 2) : 0 > n ? (x = V + ga \* Math.sin(R) - B - 3 + 2, H += -ga \* Math.cos(R) - Ja \* Math.sin(R) - 4) : x -= B + u + 3 + 3, x -= oa) : (0 < n ? (x = V + ga / 2 \* Math.sin(R) + B + 3, H -= ga / 2 \* Math.cos(R)) : 0 > n ? (x = V + B + 3 - ga / 2 \* Math.sin(R) + 2, H += ga / 2 \* Math.cos(R)) : x += B + v + 3 + 3, x += oa)) : (x += ka + u / 2 - ba, H += qa, P ? (0 < n ? (x = ga / 2 \* Math.sin(R) + B + 3, H -= ga / 2 \* Math.cos(R)) : x += B + 3, x += oa) : (0 < n ? (x = -(ga / 2) \* Math.sin(R) - B - 6, H += ga / 2 \* Math.cos(R)) : x -= B + u + 3 + v + 3, x -= oa)), "bottom" == ca ? X = (P ? V - B - 1 : V + v - 1) + ba : (la = qa, X = (P ? ka : ka - B - v + 1) - ba), f && (H += f), f = H, 0 < n &&

(f += ga / 2 \* Math.cos(R)), Z && (u = 0, P && (u = ga / 2 \* Math.cos(R)), f + u > sa + 2 || 0 > f)) Z.remove(), Z = null

} else {

0 <= b && b <= V + 1 && (0 < B && 0 < A && b + l <= V + 1 && (da = d.line(z, [0, B], [b + l, b + l], w, A, D), y.push(da)), 0 < C && (ta = d.line(z, [0, qa, sa + qa], [b, b + ka, b + ka], J, C, D, E), y.push(ta)));

fa = "end";

if (!0 === P && "left" == ca || !1 === P && "right" == ca) fa = "start";

x = b - Ja / 2 + 2;

1 == ea && 0 < I && !g && !m && (e = d.fitToBounds(b, 0, V), aa = d.fitToBounds(aa, 0, V), R = e - aa, wa = d.polygon(z, [0, a.width, a.width, 0], [0, 0, R, R], O, I), wa.translate(qa, e - R + ka), y.push(wa));

x += u / 2;

"right" == ca ? (H +=

qa + sa + ba, x += ka, P ? (f || (x -= u / 2 + 3), H = H - (B + 4) - oa) : (H += B + 4 + v, x -= 2, H += oa)) : P ? (H += B + 4 - ba, f || (x -= u / 2 + 3), g && (H += qa, x += ka), H += oa) : (H += -B - v - 4 - 2 - ba, x -= 2, H -= oa);

da && ("right" == ca ? (la += qa + ba + sa, X += ka, la = P ? la - v : la + v) : (la -= ba, P || (la -= B + v)));

f && (x += f);

P = -3;

"right" == ca && (P += ka);

Z && (x > V + 1 || x < P) && (Z.remove(), Z = null)

}

da && (da.translate(la, X), d.setCN(q, da, a.bcn + "tick"), d.setCN(q, da, p, !0), g && d.setCN(q, da, "guide"));

!1 === a.visible && (da && da.remove(), Z && (Z.remove(), Z = null));

Z && (Z.attr({"text-anchor": fa}), Z.translate(H, x), 0 !==

n && Z.rotate(-n, a.chart.backgroundColor), a.allLabels.push(Z), this.label = Z, d.setCN(q, Z, a.bcn + "label"), d.setCN(q, Z, p, !0), g && d.setCN(q, Z, "guide"));

ta && (d.setCN(q, ta, a.bcn + "grid"), d.setCN(q, ta, p, !0), g && d.setCN(q, ta, "guide"));

wa && (d.setCN(q, wa, a.bcn + "fill"), d.setCN(q, wa, p, !0));

m ? ta && d.setCN(q, ta, a.bcn + "grid-minor") : (a.counter = 0 === ea ? 1 : 0, a.previousCoord = b);

0 === this.set.node.childNodes.length && this.set.remove()

}, graphics: function () {

return this.set

}, getLabel: function () {

return this.label

}

})

})();

(function () {

var d = window.AmCharts;

d.RecFill = d.Class({

construct: function (a, b, c, e) {

var h = a.dx, f = a.dy, g = a.orientation, k = 0;

if (c < b) {

var l = b;

b = c;

c = l

}

var m = e.fillAlpha;

isNaN(m) && (m = 0);

var l = a.chart.container, n = e.fillColor;

"V" == g ? (b = d.fitToBounds(b, 0, a.viH), c = d.fitToBounds(c, 0, a.viH)) : (b = d.fitToBounds(b, 0, a.viW), c = d.fitToBounds(c, 0, a.viW));

c -= b;

isNaN(c) && (c = 4, k = 2, m = 0);

0 > c && "object" == typeof n && (n = n.join(",").split(",").reverse());

"V" == g ? (g = d.rect(l, a.viW, c, n, m), g.translate(h, b - k + f)) : (g = d.rect(l, c, a.viH, n,

m), g.translate(b - k + h, f));

d.setCN(a.chart, g, "guide-fill");

e.id && d.setCN(a.chart, g, "guide-fill-" + e.id);

this.set = l.set([g])

}, graphics: function () {

return this.set

}, getLabel: function () {

}

})

})();

(function () {

var d = window.AmCharts;

d.AmChart = d.Class({

construct: function (a) {

this.theme = a;

this.classNamePrefix = "amcharts";

this.addClassNames = !1;

this.version = "3.15.2";

d.addChart(this);

this.createEvents("dataUpdated", "init", "rendered", "drawn", "failed", "resized");

this.height = this.width = "100%";

this.dataChanged = !0;

this.chartCreated = !1;

this.previousWidth = this.previousHeight = 0;

this.backgroundColor = "#FFFFFF";

this.borderAlpha = this.backgroundAlpha = 0;

this.color = this.borderColor = "#000000";

this.fontFamily = "Verdana";

this.fontSize = 11;

this.usePrefixes = !1;

this.addCodeCredits = this.autoResize = !0;

this.precision = -1;

this.percentPrecision = 2;

this.decimalSeparator = ".";

this.thousandsSeparator = ",";

this.labels = [];

this.allLabels = [];

this.titles = [];

this.marginRight = this.marginLeft = this.autoMarginOffset = 0;

this.timeOuts = [];

this.creditsPosition = "top-left";

var b = document.createElement("div"), c = b.style;

c.overflow = "hidden";

c.position = "relative";

c.textAlign = "left";

this.chartDiv = b;

b = document.createElement("div");

c = b.style;

c.overflow =

"hidden";

c.position = "relative";

c.textAlign = "left";

this.legendDiv = b;

this.titleHeight = 0;

this.hideBalloonTime = 150;

this.handDrawScatter = 2;

this.handDrawThickness = 1;

this.prefixesOfBigNumbers = [{number: 1E3, prefix: "k"}, {number: 1E6, prefix: "M"}, {

number: 1E9,

prefix: "G"

}, {number: 1E12, prefix: "T"}, {number: 1E15, prefix: "P"}, {number: 1E18, prefix: "E"}, {

number: 1E21,

prefix: "Z"

}, {number: 1E24, prefix: "Y"}];

this.prefixesOfSmallNumbers = [{number: 1E-24, prefix: "y"}, {number: 1E-21, prefix: "z"}, {

number: 1E-18,

prefix: "a"

}, {

number: 1E-15,

prefix: "f"

}, {number: 1E-12, prefix: "p"}, {number: 1E-9, prefix: "n"}, {

number: 1E-6,

prefix: "\u03bc"

}, {number: .001, prefix: "m"}];

this.panEventsEnabled = !0;

this.product = "amcharts";

this.animations = [];

this.balloon = new d.AmBalloon(this.theme);

this.balloon.chart = this;

d.applyTheme(this, a, "AmChart")

}, drawChart: function () {

0 < this.realWidth && 0 < this.realHeight && (this.drawBackground(), this.redrawLabels(), this.drawTitles(), this.brr(), this.renderFix())

}, drawBackground: function () {

d.remove(this.background);

var a = this.container,

b = this.backgroundColor, c = this.backgroundAlpha, e = this.set;

d.isModern || 0 !== c || (c = .001);

var h = this.updateWidth();

this.realWidth = h;

var f = this.updateHeight();

this.realHeight = f;

b = d.polygon(a, [0, h - 1, h - 1, 0], [0, 0, f - 1, f - 1], b, c, 1, this.borderColor, this.borderAlpha);

d.setCN(this, b, "bg");

this.background = b;

e.push(b);

if (b = this.backgroundImage) a = a.image(b, 0, 0, h, f), d.setCN(this, b, "bg-image"), this.bgImg = a, e.push(a)

}, drawTitles: function (a) {

var b = this.titles;

this.titleHeight = 0;

if (d.ifArray(b)) {

var c = 20, e;

for (e = 0; e < b.length; e++) {

var h =

b[e], h = d.processObject(h, d.Title, this.theme);

if (!1 !== h.enabled) {

var f = h.color;

void 0 === f && (f = this.color);

var g = h.size;

isNaN(g) && (g = this.fontSize + 2);

isNaN(h.alpha);

var k = this.marginLeft,

f = d.wrappedText(this.container, h.text, f, this.fontFamily, g, "middle", !0, this.divRealWidth - 20);

f.translate(k + (this.realWidth - this.marginRight - k) / 2, c);

f.node.style.pointerEvents = "none";

h.sprite = f;

d.setCN(this, f, "title");

h.id && d.setCN(this, f, "title-" + h.id);

k = !0;

void 0 !== h.bold && (k = h.bold);

k && f.attr({"font-weight": "bold"});

f.attr({opacity: h.alpha});

c += f.getBBox().height + 5;

a ? f.remove() : this.freeLabelsSet.push(f)

}

}

this.titleHeight = c - 10

}

}, write: function (a) {

if (a = "object" != typeof a ? document.getElementById(a) : a) {

for (; a.firstChild;) a.removeChild(a.firstChild);

this.div = a;

a.style.overflow = "hidden";

a.style.textAlign = "left";

var b = this.chartDiv, c = this.legendDiv, e = this.legend, h = c.style, f = b.style;

this.measure();

this.previousHeight = this.divRealHeight;

this.previousWidth = this.divRealWidth;

var g, k = document.createElement("div");

g = k.style;

g.position = "relative";

this.containerDiv = k;

k.className = this.classNamePrefix + "-main-div";

b.className = this.classNamePrefix + "-chart-div";

a.appendChild(k);

var l = this.exportConfig;

l && d.AmExport && !this.AmExport && (this.AmExport = new d.AmExport(this, l));

this.amExport && d.AmExport && (this.AmExport = d.extend(this.amExport, new d.AmExport(this), !0));

this.AmExport && this.AmExport.init && this.AmExport.init();

if (e) if (e = this.addLegend(e, e.divId), e.enabled) switch (e.position) {

case "bottom":

k.appendChild(b);

k.appendChild(c);

break;

case "top":

k.appendChild(c);

k.appendChild(b);

break;

case "absolute":

g.width = a.style.width;

g.height = a.style.height;

h.position = "absolute";

f.position = "absolute";

void 0 !== e.left && (h.left = e.left + "px");

void 0 !== e.right && (h.right = e.right + "px");

void 0 !== e.top && (h.top = e.top + "px");

void 0 !== e.bottom && (h.bottom = e.bottom + "px");

e.marginLeft = 0;

e.marginRight = 0;

k.appendChild(b);

k.appendChild(c);

break;

case "right":

g.width = a.style.width;

g.height = a.style.height;

h.position = "relative";

f.position = "absolute";

k.appendChild(b);

k.appendChild(c);

break;

case "left":

g.width = a.style.width;

g.height = a.style.height;

h.position = "absolute";

f.position = "relative";

k.appendChild(b);

k.appendChild(c);

break;

case "outside":

k.appendChild(b)

} else k.appendChild(b); else k.appendChild(b);

this.listenersAdded || (this.addListeners(), this.listenersAdded = !0);

this.initChart()

}

}, createLabelsSet: function () {

d.remove(this.labelsSet);

this.labelsSet = this.container.set();

this.freeLabelsSet.push(this.labelsSet)

}, initChart: function () {

window.AmCharts\_path && (this.path =

window.AmCharts\_path);

void 0 === this.path && (this.path = d.getPath());

void 0 === this.path && (this.path = "amcharts/");

this.path = d.normalizeUrl(this.path);

void 0 === this.pathToImages && (this.pathToImages = this.path + "images/");

this.initHC || (d.callInitHandler(this), this.initHC = !0);

d.applyLang(this.language, this);

var a = this.numberFormatter;

a && (isNaN(a.precision) || (this.precision = a.precision), void 0 !== a.thousandsSeparator && (this.thousandsSeparator = a.thousandsSeparator), void 0 !== a.decimalSeparator && (this.decimalSeparator =

a.decimalSeparator));

(a = this.percentFormatter) && !isNaN(a.precision) && (this.percentPrecision = a.precision);

this.nf = {

precision: this.precision,

thousandsSeparator: this.thousandsSeparator,

decimalSeparator: this.decimalSeparator

};

this.pf = {

precision: this.percentPrecision,

thousandsSeparator: this.thousandsSeparator,

decimalSeparator: this.decimalSeparator

};

this.destroy();

(a = this.container) ? (a.container.innerHTML = "", a.width = this.realWidth, a.height = this.realHeight, a.addDefs(this), this.chartDiv.appendChild(a.container)) :

a = new d.AmDraw(this.chartDiv, this.realWidth, this.realHeight, this);

this.container = a;

this.checkDisplay();

a.chart = this;

d.VML || d.SVG ? (a.handDrawn = this.handDrawn, a.handDrawScatter = this.handDrawScatter, a.handDrawThickness = this.handDrawThickness, this.set && this.set.remove(), this.set = a.set(), this.gridSet && this.gridSet.remove(), this.gridSet = a.set(), this.cursorLineSet && this.cursorLineSet.remove(), this.cursorLineSet = a.set(), this.graphsBehindSet && this.graphsBehindSet.remove(), this.graphsBehindSet = a.set(), this.bulletBehindSet &&

this.bulletBehindSet.remove(), this.bulletBehindSet = a.set(), this.columnSet && this.columnSet.remove(), this.columnSet = a.set(), this.graphsSet && this.graphsSet.remove(), this.graphsSet = a.set(), this.trendLinesSet && this.trendLinesSet.remove(), this.trendLinesSet = a.set(), this.axesSet && this.axesSet.remove(), this.axesSet = a.set(), this.cursorSet && this.cursorSet.remove(), this.cursorSet = a.set(), this.scrollbarsSet && this.scrollbarsSet.remove(), this.scrollbarsSet = a.set(), this.bulletSet && this.bulletSet.remove(), this.bulletSet =

a.set(), this.freeLabelsSet && this.freeLabelsSet.remove(), this.axesLabelsSet && this.axesLabelsSet.remove(), this.axesLabelsSet = a.set(), this.freeLabelsSet = a.set(), this.balloonsSet && this.balloonsSet.remove(), this.balloonsSet = a.set(), this.zoomButtonSet && this.zoomButtonSet.remove(), this.zbSet && (this.zbSet.remove(), this.zbSet = null), this.zoomButtonSet = a.set(), this.linkSet && this.linkSet.remove(), this.linkSet = a.set()) : this.fire("failed", {

type: "failed",

chart: this

})

}, premeasure: function () {

var a = this.div;

if (a) {

this.boundingRect =

this.chartDiv.getBoundingClientRect();

var b = a.offsetWidth, c = a.offsetHeight;

a.clientHeight && (b = a.clientWidth, c = a.clientHeight);

if (b != this.mw || c != this.mh) this.mw = b, this.mh = c, this.measure()

}

}, measure: function () {

var a = this.div;

if (a) {

var b = this.chartDiv, c = a.offsetWidth, e = a.offsetHeight, h = this.container;

a.clientHeight && (c = a.clientWidth, e = a.clientHeight);

var f = d.removePx(d.getStyle(a, "padding-left")), g = d.removePx(d.getStyle(a, "padding-right")),

k = d.removePx(d.getStyle(a, "padding-top")), l = d.removePx(d.getStyle(a,

"padding-bottom"));

isNaN(f) || (c -= f);

isNaN(g) || (c -= g);

isNaN(k) || (e -= k);

isNaN(l) || (e -= l);

f = a.style;

a = f.width;

f = f.height;

-1 != a.indexOf("px") && (c = d.removePx(a));

-1 != f.indexOf("px") && (e = d.removePx(f));

e = Math.round(e);

c = Math.round(c);

a = Math.round(d.toCoordinate(this.width, c));

f = Math.round(d.toCoordinate(this.height, e));

(c != this.previousWidth || e != this.previousHeight) && 0 < a && 0 < f && (b.style.width = a + "px", b.style.height = f + "px", b.style.padding = 0, h && h.setSize(a, f), this.balloon = d.processObject(this.balloon, d.AmBalloon,

this.theme), this.balloon.setBounds(2, 2, a - 2, f));

this.balloon.chart = this;

this.realWidth = a;

this.realHeight = f;

this.divRealWidth = c;

this.divRealHeight = e

}

}, checkDisplay: function () {

if (this.container) {

var a = d.rect(this.container, 10, 10), b = a.getBBox();

0 === b.width && 0 === b.height && (this.divRealHeight = this.divRealWidth = this.realHeight = this.realWidth = 0, this.previousWidth = this.previousHeight = NaN);

a.remove()

}

}, destroy: function () {

this.chartDiv.innerHTML = "";

this.clearTimeOuts();

this.legend && this.legend.destroy()

}, clearTimeOuts: function () {

var a =

this.timeOuts;

if (a) {

var b;

for (b = 0; b < a.length; b++) clearTimeout(a[b])

}

this.timeOuts = []

}, clear: function (a) {

d.callMethod("clear", [this.chartScrollbar, this.scrollbarV, this.scrollbarH, this.chartCursor]);

this.chartCursor = this.scrollbarH = this.scrollbarV = this.chartScrollbar = null;

this.clearTimeOuts();

this.container && (this.container.remove(this.chartDiv), this.container.remove(this.legendDiv));

a || d.removeChart(this);

if (a = this.div) for (; a.firstChild;) a.removeChild(a.firstChild);

this.legend && this.legend.destroy()

},

setMouseCursor: function (a) {

"auto" == a && d.isNN && (a = "default");

this.chartDiv.style.cursor = a;

this.legendDiv.style.cursor = a

}, redrawLabels: function () {

this.labels = [];

var a = this.allLabels;

this.createLabelsSet();

var b;

for (b = 0; b < a.length; b++) this.drawLabel(a[b])

}, drawLabel: function (a) {

if (this.container && !1 !== a.enabled) {

a = d.processObject(a, d.Label, this.theme);

var b = a.y, c = a.text, e = a.align, h = a.size, f = a.color, g = a.rotation, k = a.alpha, l = a.bold,

m = d.toCoordinate(a.x, this.realWidth), b = d.toCoordinate(b, this.realHeight);

m || (m = 0);

b || (b = 0);

void 0 === f && (f = this.color);

isNaN(h) && (h = this.fontSize);

e || (e = "start");

"left" == e && (e = "start");

"right" == e && (e = "end");

"center" == e && (e = "middle", g ? b = this.realHeight - b + b / 2 : m = this.realWidth / 2 - m);

void 0 === k && (k = 1);

void 0 === g && (g = 0);

b += h / 2;

c = d.text(this.container, c, f, this.fontFamily, h, e, l, k);

c.translate(m, b);

d.setCN(this, c, "label");

a.id && d.setCN(this, c, "label-" + a.id);

0 !== g && c.rotate(g);

a.url ? (c.setAttr("cursor", "pointer"), c.click(function () {

d.getURL(a.url)

})) : c.node.style.pointerEvents = "none";

this.labelsSet.push(c);

this.labels.push(c)

}

}, addLabel: function (a, b, c, e, d, f, g, k, l, m) {

a = {x: a, y: b, text: c, align: e, size: d, color: f, alpha: k, rotation: g, bold: l, url: m, enabled: !0};

this.container && this.drawLabel(a);

this.allLabels.push(a)

}, clearLabels: function () {

var a = this.labels, b;

for (b = a.length - 1; 0 <= b; b--) a[b].remove();

this.labels = [];

this.allLabels = []

}, updateHeight: function () {

var a = this.divRealHeight, b = this.legend;

if (b) {

var c = this.legendDiv.offsetHeight, b = b.position;

if ("top" == b || "bottom" == b) {

a -= c;

if (0 > a || isNaN(a)) a =

0;

this.chartDiv.style.height = a + "px"

}

}

return a

}, updateWidth: function () {

var a = this.divRealWidth, b = this.divRealHeight, c = this.legend;

if (c) {

var e = this.legendDiv, d = e.offsetWidth;

isNaN(c.width) || (d = c.width);

c.ieW && (d = c.ieW);

var f = e.offsetHeight, e = e.style, g = this.chartDiv.style, c = c.position;

if ("right" == c || "left" == c) {

a -= d;

if (0 > a || isNaN(a)) a = 0;

g.width = a + "px";

"left" == c ? (g.left = d + "px", e.left = "0px") : (g.left = "0px", e.left = a + "px");

b > f && (e.top = (b - f) / 2 + "px")

}

}

return a

}, getTitleHeight: function () {

this.drawTitles(!0);

return this.titleHeight

}, addTitle: function (a, b, c, e, d) {

isNaN(b) && (b = this.fontSize + 2);

a = {text: a, size: b, color: c, alpha: e, bold: d, enabled: !0};

this.titles.push(a);

return a

}, handleWheel: function (a) {

var b = 0;

a || (a = window.event);

a.wheelDelta ? b = a.wheelDelta / 120 : a.detail && (b = -a.detail / 3);

b && this.handleWheelReal(b, a.shiftKey);

a.preventDefault && a.preventDefault()

}, handleWheelReal: function () {

}, addListeners: function () {

var a = this, b = a.chartDiv;

document.addEventListener ? (a.panEventsEnabled && (b.style.msTouchAction = "none"),

"ontouchstart" in document.documentElement && (b.addEventListener("touchstart", function (b) {

a.handleTouchStart.call(a, b)

}, !0), b.addEventListener("touchmove", function (b) {

a.handleMouseMove.call(a, b)

}, !0), b.addEventListener("touchend", function (b) {

a.handleTouchEnd.call(a, b)

}, !0)), b.addEventListener("mousedown", function (b) {

a.mouseIsOver = !0;

a.handleMouseMove.call(a, b);

a.handleMouseDown.call(a, b)

}, !0), b.addEventListener("mouseover", function (b) {

a.handleMouseOver.call(a, b)

}, !0), b.addEventListener("mouseout", function (b) {

a.handleMouseOut.call(a,

b)

}, !0)) : (b.attachEvent("onmousedown", function (b) {

a.handleMouseDown.call(a, b)

}), b.attachEvent("onmouseover", function (b) {

a.handleMouseOver.call(a, b)

}), b.attachEvent("onmouseout", function (b) {

a.handleMouseOut.call(a, b)

}))

}, dispDUpd: function () {

if (!this.skipEvents) {

var a;

this.dispatchDataUpdated && (this.dispatchDataUpdated = !1, a = "dataUpdated", this.fire(a, {

type: a,

chart: this

}));

this.chartCreated || (this.chartCreated = !0, a = "init", this.fire(a, {type: a, chart: this}));

this.chartRendered || (a = "rendered", this.fire(a,

{type: a, chart: this}), this.chartRendered = !0);

a = "drawn";

this.fire(a, {type: a, chart: this})

}

this.skipEvents = !1

}, validateSize: function () {

var a = this;

a.premeasure();

a.checkDisplay();

if (a.divRealWidth != a.previousWidth || a.divRealHeight != a.previousHeight) {

var b = a.legend;

if (0 < a.realWidth && 0 < a.realHeight) {

a.sizeChanged = !0;

if (b) {

a.legendInitTO && clearTimeout(a.legendInitTO);

var c = setTimeout(function () {

b.invalidateSize()

}, 10);

a.timeOuts.push(c);

a.legendInitTO = c

}

"xy" != a.type ? a.marginsUpdated = !1 : (a.marginsUpdated =

!0, a.selfZoom = !0);

clearTimeout(a.initTO);

c = setTimeout(function () {

a.initChart()

}, 10);

a.timeOuts.push(c);

a.initTO = c

}

a.fire("resized", {type: "resized", chart: a});

a.renderFix();

b && b.renderFix && b.renderFix();

a.previousHeight = a.divRealHeight;

a.previousWidth = a.divRealWidth

}

}, invalidateSize: function () {

this.previousHeight = this.previousWidth = NaN;

this.invalidateSizeReal()

}, invalidateSizeReal: function () {

var a = this;

a.marginsUpdated = !1;

clearTimeout(a.validateTO);

var b = setTimeout(function () {

a.validateSize()

}, 5);

a.timeOuts.push(b);

a.validateTO = b

}, validateData: function (a) {

this.chartCreated && (this.dataChanged = !0, this.marginsUpdated = !1, this.initChart(a))

}, validateNow: function (a, b) {

this.initTO && clearTimeout(this.initTO);

a && (this.dataChanged = !0);

this.skipEvents = b;

this.chartRendered = !1;

this.write(this.div)

}, showItem: function (a) {

a.hidden = !1;

this.initChart()

}, hideItem: function (a) {

a.hidden = !0;

this.initChart()

}, hideBalloon: function () {

var a = this;

clearTimeout(a.hoverInt);

clearTimeout(a.balloonTO);

a.hoverInt = setTimeout(function () {

a.hideBalloonReal.call(a)

},

a.hideBalloonTime)

}, cleanChart: function () {

}, hideBalloonReal: function () {

var a = this.balloon;

a && a.hide()

}, showBalloon: function (a, b, c, e, d) {

var f = this;

clearTimeout(f.balloonTO);

clearTimeout(f.hoverInt);

f.balloonTO = setTimeout(function () {

f.showBalloonReal.call(f, a, b, c, e, d)

}, 1)

}, showBalloonReal: function (a, b, c, e, d) {

this.handleMouseMove();

var f = this.balloon;

f.enabled && (f.followCursor(!1), f.changeColor(b), !c || f.fixedPosition ? (f.setPosition(e, d), f.followCursor(!1)) : f.followCursor(!0), a && f.showBalloon(a))

}, handleMouseOver: function () {

this.outTO &&

clearTimeout(this.outTO);

d.resetMouseOver();

this.mouseIsOver = !0

}, handleMouseOut: function () {

var a = this;

a.outTO && clearTimeout(a.outTO);

a.outTO = setTimeout(function () {

a.handleMouseOutReal()

}, 10)

}, handleMouseOutReal: function () {

d.resetMouseOver();

this.mouseIsOver = !1

}, handleMouseMove: function (a) {

a || (a = window.event);

if (a) {

if (a.touches && (a = a.touches.item(0), !a)) return;

this.boundingRect && a.clientX && (this.mouseX = a.clientX - this.boundingRect.left, this.mouseY = a.clientY - this.boundingRect.top)

}

}, handleTouchStart: function (a) {

this.handleMouseMove(a);

this.handleMouseDown(a)

}, handleTouchEnd: function (a) {

this.handleMouseMove(a);

d.resetMouseOver();

this.handleReleaseOutside(a)

}, handleReleaseOutside: function () {

}, handleMouseDown: function (a) {

d.resetMouseOver();

this.mouseIsOver = !0;

a && a.preventDefault && (this.panEventsEnabled ? a.preventDefault() : a.touches || a.preventDefault())

}, addLegend: function (a, b) {

a = d.processObject(a, d.AmLegend, this.theme);

a.divId = b;

a.ieW = 0;

var c;

c = "object" != typeof b && b ? document.getElementById(b) : b;

this.legend = a;

a.chart = this;

c ? (a.div =

c, a.position = "outside", a.autoMargins = !1) : a.div = this.legendDiv;

return a

}, removeLegend: function () {

this.legend = void 0;

this.legendDiv.innerHTML = ""

}, handleResize: function () {

(d.isPercents(this.width) || d.isPercents(this.height)) && this.invalidateSizeReal();

this.renderFix()

}, renderFix: function () {

if (!d.VML) {

var a = this.container;

a && a.renderFix()

}

}, getSVG: function () {

if (d.hasSVG) return this.container

}, animate: function (a, b, c, e, h, f, g) {

a["an\_" + b] && d.removeFromArray(this.animations, a["an\_" + b]);

c = {

obj: a, frame: 0, attribute: b,

from: c, to: e, time: h, effect: f, suffix: g

};

a["an\_" + b] = c;

this.animations.push(c);

return c

}, setLegendData: function (a) {

var b = this.legend;

b && b.setData(a)

}, stopAnim: function (a) {

d.removeFromArray(this.animations, a)

}, updateAnimations: function () {

var a;

this.container && this.container.update();

if (this.animations) for (a = this.animations.length - 1; 0 <= a; a--) {

var b = this.animations[a], c = 1E3 \* b.time / d.updateRate, e = b.frame + 1, h = b.obj,

f = b.attribute;

if (e <= c) {

b.frame++;

var g = Number(b.from), k = Number(b.to) - g, c = d[b.effect](0, e, g,

k, c);

0 === k ? (this.animations.splice(a, 1), h.node.style[f] = Number(b.to) + b.suffix) : h.node.style[f] = c + b.suffix

} else h.node.style[f] = Number(b.to) + b.suffix, this.animations.splice(a, 1)

}

}, update: function () {

this.updateAnimations()

}, inIframe: function () {

try {

return window.self !== window.top

} catch (a) {

return !0

}

}, brr: function () {

var a = "amcharts.com", b = window.location.hostname.split("."), c;

2 <= b.length && (c = b[b.length - 2] + "." + b[b.length - 1]);

this.amLink && (b = this.amLink.parentNode) && b.removeChild(this.amLink);

b = this.creditsPosition;

if (c != a || !0 === this.inIframe()) {

var a = "http://www." + a, e = c = 0, d = this.realWidth, f = this.realHeight, g = this.type;

if ("serial" == g || "xy" == g || "gantt" == g) c = this.marginLeftReal, e = this.marginTopReal, d = c + this.plotAreaWidth, f = e + this.plotAreaHeight;

var g = a + "/javascript-charts/", k = "JavaScript charts", l = "JS chart by amCharts";

"ammap" == this.product && (g = a + "/javascript-maps/", k = "Interactive JavaScript maps", l = "JS map by amCharts");

a = document.createElement("a");

l = document.createTextNode(l);

a.setAttribute("href", g);

a.setAttribute("title",

k);

a.appendChild(l);

this.chartDiv.appendChild(a);

this.amLink = a;

g = a.style;

g.position = "absolute";

g.textDecoration = "none";

g.color = this.color;

g.fontFamily = this.fontFamily;

g.fontSize = this.fontSize + "px";

g.opacity = .7;

g.display = "block";

var k = a.offsetWidth, a = a.offsetHeight, l = 5 + c, m = e + 5;

"bottom-left" == b && (l = 5 + c, m = f - a - 3);

"bottom-right" == b && (l = d - k - 5, m = f - a - 3);

"top-right" == b && (l = d - k - 5, m = e + 5);

g.left = l + "px";

g.top = m + "px"

}

}

});

d.Slice = d.Class({

construct: function () {

}

});

d.SerialDataItem = d.Class({

construct: function () {

}

});

d.GraphDataItem = d.Class({

construct: function () {

}

});

d.Guide = d.Class({

construct: function (a) {

this.cname = "Guide";

d.applyTheme(this, a, this.cname)

}

});

d.Title = d.Class({

construct: function (a) {

this.cname = "Title";

d.applyTheme(this, a, this.cname)

}

});

d.Label = d.Class({

construct: function (a) {

this.cname = "Label";

d.applyTheme(this, a, this.cname)

}

})

})();

(function () {

var d = window.AmCharts;

d.AmGraph = d.Class({

construct: function (a) {

this.cname = "AmGraph";

this.createEvents("rollOverGraphItem", "rollOutGraphItem", "clickGraphItem", "doubleClickGraphItem", "rightClickGraphItem", "clickGraph", "rollOverGraph", "rollOutGraph");

this.type = "line";

this.stackable = !0;

this.columnCount = 1;

this.columnIndex = 0;

this.centerCustomBullets = this.showBalloon = !0;

this.maxBulletSize = 50;

this.minBulletSize = 4;

this.balloonText = "[[value]]";

this.hidden = this.scrollbar = this.animationPlayed = !1;

this.pointPosition = "middle";

this.depthCount = 1;

this.includeInMinMax = !0;

this.negativeBase = 0;

this.visibleInLegend = !0;

this.showAllValueLabels = !1;

this.showBulletsAt = this.showBalloonAt = "close";

this.lineThickness = 1;

this.dashLength = 0;

this.connect = !0;

this.lineAlpha = 1;

this.bullet = "none";

this.bulletBorderThickness = 2;

this.bulletBorderAlpha = 0;

this.bulletAlpha = 1;

this.bulletSize = 8;

this.cornerRadiusTop = this.hideBulletsCount = this.bulletOffset = 0;

this.cursorBulletAlpha = 1;

this.gradientOrientation = "vertical";

this.dy =

this.dx = 0;

this.periodValue = "";

this.clustered = !0;

this.periodSpan = 1;

this.y = this.x = 0;

this.switchable = !0;

this.tcc = this.minDistance = 1;

this.labelRotation = 0;

this.labelAnchor = "auto";

this.labelOffset = 3;

this.bcn = "graph-";

this.dateFormat = "MMM DD, YYYY";

d.applyTheme(this, a, this.cname)

}, draw: function () {

var a = this.chart, b = a.type;

isNaN(this.precision) || (this.numberFormatter ? this.numberFormatter.precision = this.precision : this.numberFormatter = {

precision: this.precision,

decimalSeparator: a.decimalSeparator,

thousandsSeparator: a.thousandsSeparator

});

var c = a.container;

this.container = c;

this.destroy();

var e = c.set(), h = c.set();

this.behindColumns ? (a.graphsBehindSet.push(e), a.bulletBehindSet.push(h)) : (a.graphsSet.push(e), a.bulletSet.push(h));

var f = this.bulletAxis;

d.isString(f) && (this.bulletAxis = a.getValueAxisById(f));

this.bulletSet = h;

if (!this.scrollbar) {

var f = a.marginLeftReal, g = a.marginTopReal;

e.translate(f, g);

h.translate(f, g)

}

c = c.set();

d.remove(this.columnsSet);

e.push(c);

this.set = e;

d.setCN(a, e, "graph-" + this.type);

d.setCN(a, e, "graph-" + this.id);

d.setCN(a,

h, "graph-" + this.type);

d.setCN(a, h, "graph-" + this.id);

this.columnsSet = c;

this.columnsArray = [];

this.ownColumns = [];

this.allBullets = [];

this.animationArray = [];

e = this.labelPosition;

e || (h = this.valueAxis.stackType, e = "top", "column" == this.type && (a.rotate && (e = "right"), "100%" == h || "regular" == h) && (e = "middle"), this.labelPosition = e);

d.ifArray(this.data) && (a = !1, "xy" == b ? this.xAxis.axisCreated && this.yAxis.axisCreated && (a = !0) : this.valueAxis.axisCreated && (a = !0), !this.hidden && a && this.createGraph())

}, createGraph: function () {

var a =

this, b = a.chart;

a.startAlpha = b.startAlpha;

a.seqAn = b.sequencedAnimation;

a.baseCoord = a.valueAxis.baseCoord;

void 0 === a.fillAlphas && (a.fillAlphas = 0);

a.bulletColorR = a.bulletColor;

void 0 === a.bulletColorR && (a.bulletColorR = a.lineColorR, a.bulletColorNegative = a.negativeLineColor);

void 0 === a.bulletAlpha && (a.bulletAlpha = a.lineAlpha);

b = b.type;

"gantt" == b && (b = "serial");

clearTimeout(a.playedTO);

if (!isNaN(a.valueAxis.min) && !isNaN(a.valueAxis.max)) {

switch (b) {

case "serial":

a.categoryAxis && (a.createSerialGraph(), "candlestick" ==

a.type && 1 > a.valueAxis.minMaxMultiplier && a.positiveClip(a.set));

break;

case "radar":

a.createRadarGraph();

break;

case "xy":

a.createXYGraph(), a.positiveClip(a.set)

}

a.playedTO = setTimeout(function () {

a.setAnimationPlayed.call(a)

}, 500 \* a.chart.startDuration)

}

}, setAnimationPlayed: function () {

this.animationPlayed = !0

}, createXYGraph: function () {

var a = [], b = [], c = this.xAxis, e = this.yAxis;

this.pmh = e.viH + 1;

this.pmw = c.viW + 1;

this.pmy = this.pmx = 0;

var d;

for (d = this.start; d <= this.end; d++) {

var f = this.data[d].axes[c.id].graphs[this.id],

g = f.values, k = g.x, l = g.y, g = c.getCoordinate(k), m = e.getCoordinate(l);

!isNaN(k) && !isNaN(l) && (a.push(g), b.push(m), f.x = g, f.y = m, k = this.createBullet(f, g, m, d), l = this.labelText) && (f = this.createLabel(f, g, m, l), this.positionLabel(g, m, f, k), this.allBullets.push(f))

}

this.drawLineGraph(a, b);

this.launchAnimation()

}, createRadarGraph: function () {

var a = this.valueAxis.stackType, b = [], c = [], e = [], d = [], f, g, k, l, m;

for (m = this.start; m <= this.end; m++) {

var n = this.data[m].axes[this.valueAxis.id].graphs[this.id], p, q;

"none" == a || "3d" ==

a ? p = n.values.value : (p = n.values.close, q = n.values.open);

if (isNaN(p)) this.connect || (this.drawLineGraph(b, c, e, d), b = [], c = [], e = [], d = []); else {

var r = this.y - (this.valueAxis.getCoordinate(p) - this.height),

r = r \* this.valueAxis.rMultiplier, u = 180 - 360 / (this.end - this.start + 1) \* m;

"middle" == this.valueAxis.pointPosition && (u -= 180 / (this.end - this.start + 1));

p = r \* Math.sin(u / 180 \* Math.PI);

r \*= Math.cos(u / 180 \* Math.PI);

b.push(p);

c.push(r);

if (!isNaN(q)) {

var t = this.y - (this.valueAxis.getCoordinate(q) - this.height),

t = t \* this.valueAxis.rMultiplier,

z = t \* Math.sin(u / 180 \* Math.PI), u = t \* Math.cos(u / 180 \* Math.PI);

e.push(z);

d.push(u);

isNaN(k) && (k = z);

isNaN(l) && (l = u)

}

u = this.createBullet(n, p, r, m);

if (z = this.labelText) n = this.createLabel(n, p, r, z), this.positionLabel(p, r, n, u), this.allBullets.push(n);

isNaN(f) && (f = p);

isNaN(g) && (g = r)

}

}

b.push(f);

c.push(g);

isNaN(k) || (e.push(k), d.push(l));

this.drawLineGraph(b, c, e, d);

this.launchAnimation()

}, positionLabel: function (a, b, c, e) {

var d = "middle", f = !1, g = this.labelPosition, k = c.getBBox();

if (e) {

var l = e.graphDataItem, m = this.chart.rotate,

n = l.isNegative, p = this.chart, q = this.valueAxis;

b -= k.height / 4 / 2;

switch (g) {

case "top":

g = m ? "top" : n ? "bottom" : "top";

break;

case "right":

g = m ? n ? "left" : "right" : "right";

break;

case "bottom":

g = m ? "bottom" : n ? "top" : "bottom";

break;

case "left":

g = m ? n ? "right" : "left" : "left"

}

var r = l.columnGraphics, u = 0, t = 0;

r && (u = r.x, t = r.y);

var z = this.labelOffset;

switch (g) {

case "top":

b = q.reversed ? b + (e.size / 2 + k.height / 2 + z) : b - (e.size / 2 + k.height / 2 + z);

break;

case "right":

d = "start";

a += e.size / 2 + z;

break;

case "bottom":

b = q.reversed ? b - (e.size / 2 + k.height /

2 + z) : b + (e.size / 2 + k.height / 2 + z);

break;

case "left":

d = "end";

a -= e.size / 2 + z;

break;

case "inside":

"column" == this.type && (f = !0, m ? n ? (d = "end", a = u - 3 - z) : (d = "start", a = u + 3 + z) : b = n ? t + 7 + z : t - 10 - z);

break;

case "middle":

"column" == this.type && (f = !0, m ? a -= (a - u) / 2 + z - 3 : b -= (b - t) / 2 + z - 3)

}

"auto" != this.labelAnchor && (d = this.labelAnchor);

c.attr({"text-anchor": d});

this.labelRotation && c.rotate(this.labelRotation);

c.translate(a, b);

k = c.getBBox();

!this.showAllValueLabels && r && f && (k.height > l.columnHeight || k.width > l.columnWidth) && (c.remove(),

c = !1);

if (c && ("serial" == p.type || "gantt" == p.type)) if (m) {

if (0 > b || b > this.height) c.remove(), c = !1

} else if (0 > a || a > this.width) c.remove(), c = !1;

return c

}

}, getGradRotation: function () {

var a = 270;

"horizontal" == this.gradientOrientation && (a = 0);

return this.gradientRotation = a

}, createSerialGraph: function () {

this.dashLengthSwitched = this.fillColorsSwitched = this.lineColorSwitched = void 0;

var a = this.chart, b = this.id, c = this.index, e = this.data, h = this.chart.container,

f = this.valueAxis, g = this.type, k = this.columnWidthReal, l = this.showBulletsAt;

isNaN(this.columnWidth) || (k = this.columnWidth);

isNaN(k) && (k = .8);

var m = this.useNegativeColorIfDown, n = this.width, p = this.height, q = this.y, r = this.rotate,

u = this.columnCount, t = d.toCoordinate(this.cornerRadiusTop, k / 2), z = this.connect, y = [], v = [],

w, A, B, C, D = this.chart.graphs.length, J, E = this.dx / this.tcc, O = this.dy / this.tcc,

I = f.stackType, W = this.start, ea = this.end, P = this.scrollbar, oa = "graph-column-";

P && (oa = "scrollbar-graph-column-");

var qa = this.categoryAxis, ka = this.baseCoord, Qa = this.negativeBase, ca = this.columnIndex,

aa = this.lineThickness, V = this.lineAlpha, sa = this.lineColorR, ba = this.dashLength, da = this.set,

ta, fa = this.getGradRotation(), R = this.chart.columnSpacing, X = qa.cellWidth, wa = (X \* k - u) / u;

R > wa && (R = wa);

var H, x, la, ga = p + 1, Ja = n + 1, Z = 0, qb = 0, rb, sb, eb, fb, tb = this.fillColorsR,

Ka = this.negativeFillColors, Da = this.negativeLineColor, Wa = this.fillAlphas,

Xa = this.negativeFillAlphas;

"object" == typeof Wa && (Wa = Wa[0]);

"object" == typeof Xa && (Xa = Xa[0]);

var gb = f.getCoordinate(f.min);

f.logarithmic && (gb = f.getCoordinate(f.minReal));

this.minCoord =

gb;

this.resetBullet && (this.bullet = "none");

if (!(P || "line" != g && "smoothedLine" != g && "step" != g || (1 == e.length && "step" != g && "none" == this.bullet && (this.bullet = "round", this.resetBullet = !0), !Ka && void 0 == Da || m))) {

var Ra = Qa;

Ra > f.max && (Ra = f.max);

Ra < f.min && (Ra = f.min);

f.logarithmic && (Ra = f.minReal);

var Ga = f.getCoordinate(Ra), Ib = f.getCoordinate(f.max);

r ? (ga = p, Ja = Math.abs(Ib - Ga) + 1, rb = p, sb = Math.abs(gb - Ga) + 1, fb = qb = 0, f.reversed ? (Z = 0, eb = Ga) : (Z = Ga, eb = 0)) : (Ja = n, ga = Math.abs(Ib - Ga) + 1, sb = n, rb = Math.abs(gb - Ga) + 1, eb = Z = 0, f.reversed ?

(fb = q, qb = Ga) : fb = Ga + 1)

}

var Ha = Math.round;

this.pmx = Ha(Z);

this.pmy = Ha(qb);

this.pmh = Ha(ga);

this.pmw = Ha(Ja);

this.nmx = Ha(eb);

this.nmy = Ha(fb);

this.nmh = Ha(rb);

this.nmw = Ha(sb);

d.isModern || (this.nmy = this.nmx = 0, this.nmh = this.height);

this.clustered || (u = 1);

k = "column" == g ? (X \* k - R \* (u - 1)) / u : X \* k;

1 > k && (k = 1);

var Jb = this.fixedColumnWidth;

isNaN(Jb) || (k = Jb);

var K;

if ("line" == g || "step" == g || "smoothedLine" == g) {

if (0 < W) {

for (K = W - 1; -1 < K; K--) if (H = e[K], x = H.axes[f.id].graphs[b], la = x.values.value, !isNaN(la)) {

W = K;

break

}

if (this.lineColorField) for (K =

W; -1 < K; K--) if (H = e[K], x = H.axes[f.id].graphs[b], x.lineColor) {

this.bulletColorSwitched = this.lineColorSwitched = x.lineColor;

break

}

if (this.fillColorsField) for (K = W; -1 < K; K--) if (H = e[K], x = H.axes[f.id].graphs[b], x.fillColors) {

this.fillColorsSwitched = x.fillColors;

break

}

if (this.dashLengthField) for (K = W; -1 < K; K--) if (H = e[K], x = H.axes[f.id].graphs[b], !isNaN(x.dashLength)) {

this.dashLengthSwitched = x.dashLength;

break

}

}

if (ea < e.length - 1) for (K = ea + 1; K < e.length; K++) if (H = e[K], x = H.axes[f.id].graphs[b], la = x.values.value, !isNaN(la)) {

ea =

K;

break

}

}

ea < e.length - 1 && ea++;

var S = [], T = [], La = !1;

if ("line" == g || "step" == g || "smoothedLine" == g) if (this.stackable && "regular" == I || "100%" == I || this.fillToGraph) La = !0;

var Kb = this.noStepRisers, hb = -1E3, ib = -1E3, jb = this.minDistance, Ma = !0, Ya = !1;

for (K = W; K <= ea; K++) {

H = e[K];

x = H.axes[f.id].graphs[b];

x.index = K;

var Za, Na = NaN;

if (m && void 0 == this.openField) for (var ub = K + 1; ub < e.length && (!e[ub] || !(Za = e[K + 1].axes[f.id].graphs[b]) || !Za.values || (Na = Za.values.value, isNaN(Na))); ub++) ;

var U, Q, M, ha, ma = NaN, G = NaN, F = NaN, N = NaN, L = NaN,

Oa = NaN, Ea = NaN, Pa = NaN, Fa = NaN, ya = NaN, za = NaN, ia = NaN, ja = NaN, Y = NaN, vb = NaN,

wb = NaN, na = NaN, pa = void 0, Ia = tb, Sa = Wa, Ba = sa, ua, xa, xb = this.proCandlesticks,

kb = this.topRadius, $a = this.pattern;

void 0 != x.pattern && ($a = x.pattern);

isNaN(x.alpha) || (Sa = x.alpha);

isNaN(x.dashLength) || (ba = x.dashLength);

var Ca = x.values;

f.recalculateToPercents && (Ca = x.percents);

if (Ca) {

Y = this.stackable && "none" != I && "3d" != I ? Ca.close : Ca.value;

if ("candlestick" == g || "ohlc" == g) Y = Ca.close, wb = Ca.low, Ea = f.getCoordinate(wb), vb = Ca.high, Fa = f.getCoordinate(vb);

na = Ca.open;

F = f.getCoordinate(Y);

isNaN(na) || (L = f.getCoordinate(na), m && (Na = na, na = L = NaN));

m && (void 0 == this.openField ? Za && (Za.isNegative = Na < Y ? !0 : !1, isNaN(Na) && (x.isNegative = !Ma)) : x.isNegative = Na > Y ? !0 : !1);

if (!P) switch (this.showBalloonAt) {

case "close":

x.y = F;

break;

case "open":

x.y = L;

break;

case "high":

x.y = Fa;

break;

case "low":

x.y = Ea

}

var ma = H.x[qa.id], Ta = this.periodSpan - 1, ra = Math.floor(X / 2) + Math.floor(Ta \* X / 2),

Aa = ra, lb = 0;

"left" == this.stepDirection && (lb = (2 \* X + Ta \* X) / 2, ma -= lb);

"center" == this.stepDirection && (lb = X /

2, ma -= lb);

"start" == this.pointPosition && (ma -= X / 2 + Math.floor(Ta \* X / 2), ra = 0, Aa = Math.floor(X) + Math.floor(Ta \* X));

"end" == this.pointPosition && (ma += X / 2 + Math.floor(Ta \* X / 2), ra = Math.floor(X) + Math.floor(Ta \* X), Aa = 0);

if (Kb) {

var yb = this.columnWidth;

isNaN(yb) || (ra \*= yb, Aa \*= yb)

}

P || (x.x = ma);

-1E5 > ma && (ma = -1E5);

ma > n + 1E5 && (ma = n + 1E5);

r ? (G = F, N = L, L = F = ma, isNaN(na) && !this.fillToGraph && (N = ka), Oa = Ea, Pa = Fa) : (N = G = ma, isNaN(na) && !this.fillToGraph && (L = ka));

if (!xb && Y < na || xb && Y < ta) x.isNegative = !0, Ka && (Ia = Ka), Xa && (Sa = Xa), void 0 != Da &&

(Ba = Da);

Ya = !1;

isNaN(Y) || (m ? Y > Na ? (Ma && (Ya = !0), Ma = !1) : (Ma || (Ya = !0), Ma = !0) : x.isNegative = Y < Qa ? !0 : !1, ta = Y);

var Lb = !1;

P && a.chartScrollbar.ignoreCustomColors && (Lb = !0);

Lb || (void 0 != x.color && (Ia = x.color), x.fillColors && (Ia = x.fillColors));

switch (g) {

case "line":

if (isNaN(Y)) z || (this.drawLineGraph(y, v, S, T), y = [], v = [], S = [], T = []); else {

if (Math.abs(G - hb) >= jb || Math.abs(F - ib) >= jb) y.push(G), v.push(F), hb = G, ib = F;

ya = G;

za = F;

ia = G;

ja = F;

!La || isNaN(L) || isNaN(N) || (S.push(N), T.push(L));

if (Ya || void 0 != x.lineColor || void 0 != x.fillColors ||

!isNaN(x.dashLength)) this.drawLineGraph(y, v, S, T), y = [G], v = [F], S = [], T = [], !La || isNaN(L) || isNaN(N) || (S.push(N), T.push(L)), m ? Ma ? (this.lineColorSwitched = sa, this.fillColorsSwitched = tb) : (this.lineColorSwitched = Da, this.fillColorsSwitched = Ka) : (this.lineColorSwitched = x.lineColor, this.fillColorsSwitched = x.fillColors), this.dashLengthSwitched = x.dashLength;

x.gap && (this.drawLineGraph(y, v, S, T), y = [], v = [], S = [], T = [])

}

break;

case "smoothedLine":

if (isNaN(Y)) z || (this.drawSmoothedGraph(y, v, S, T), y = [], v = [], S = [], T = []); else {

if (Math.abs(G -

hb) >= jb || Math.abs(F - ib) >= jb) y.push(G), v.push(F), hb = G, ib = F;

ya = G;

za = F;

ia = G;

ja = F;

!La || isNaN(L) || isNaN(N) || (S.push(N), T.push(L));

void 0 == x.lineColor && void 0 == x.fillColors && isNaN(x.dashLength) || (this.drawSmoothedGraph(y, v, S, T), y = [G], v = [F], S = [], T = [], !La || isNaN(L) || isNaN(N) || (S.push(N), T.push(L)), this.lineColorSwitched = x.lineColor, this.fillColorsSwitched = x.fillColors, this.dashLengthSwitched = x.dashLength);

x.gap && (this.drawSmoothedGraph(y, v, S, T), y = [], v = [], S = [], T = [])

}

break;

case "step":

if (!isNaN(Y)) {

r ? (isNaN(w) ||

(y.push(w), v.push(F - ra)), v.push(F - ra), y.push(G), v.push(F + Aa), y.push(G), !La || isNaN(L) || isNaN(N) || (isNaN(B) || (S.push(B), T.push(L - ra)), S.push(N), T.push(L - ra), S.push(N), T.push(L + Aa))) : (isNaN(A) || (v.push(A), y.push(G - ra)), y.push(G - ra), v.push(F), y.push(G + Aa), v.push(F), !La || isNaN(L) || isNaN(N) || (isNaN(C) || (S.push(N - ra), T.push(C)), S.push(N - ra), T.push(L), S.push(N + Aa), T.push(L)));

w = G;

A = F;

B = N;

C = L;

ya = G;

za = F;

ia = G;

ja = F;

if (Ya || void 0 != x.lineColor || void 0 != x.fillColors || !isNaN(x.dashLength)) {

var Zb = y[y.length - 2], $b =

v[v.length - 2];

y.pop();

v.pop();

this.drawLineGraph(y, v, S, T);

y = [Zb];

v = [$b];

r ? (v.push(F + Aa), y.push(G)) : (y.push(G + Aa), v.push(F));

S = [];

T = [];

this.lineColorSwitched = x.lineColor;

this.fillColorsSwitched = x.fillColors;

this.dashLengthSwitched = x.dashLength;

m && (Ma ? (this.lineColorSwitched = sa, this.fillColorsSwitched = tb) : (this.lineColorSwitched = Da, this.fillColorsSwitched = Ka))

}

if (Kb || x.gap) w = A = NaN, this.drawLineGraph(y, v, S, T), y = [], v = [], S = [], T = []

} else if (!z) {

if (1 >= this.periodSpan || 1 < this.periodSpan && G - w > ra + Aa) w = A = NaN;

this.drawLineGraph(y, v, S, T);

y = [];

v = [];

S = [];

T = []

}

break;

case "column":

ua = Ba;

void 0 != x.lineColor && (ua = x.lineColor);

if (!isNaN(Y)) {

m || (x.isNegative = Y < Qa ? !0 : !1);

x.isNegative && (Ka && (Ia = Ka), void 0 != Da && (ua = Da));

var Mb = f.min, Nb = f.max;

if (!(Y < Mb && na < Mb || Y > Nb && na > Nb)) {

var va;

if (r) {

"3d" == I ? (Q = F - (u / 2 - this.depthCount + 1) \* (k + R) + R / 2 + O \* ca, U = N + E \* ca, va = ca) : (Q = Math.floor(F - (u / 2 - ca) \* (k + R) + R / 2), U = N, va = 0);

M = k;

ya = G;

za = Q + k / 2;

isNaN(N) || N > G && !x.isNegative && (ya = N);

ia = G;

ja = Q + k / 2;

Q + M > p + va \* O && (M = p - Q + va \* O);

Q < va \* O && (M += Q, Q = va \* O);

ha =

G - N;

var ac = U;

U = d.fitToBounds(U, 0, n);

ha += ac - U;

ha = d.fitToBounds(ha, -U, n - U + E \* ca);

Q < p && 0 < M && (pa = new d.Cuboid(h, ha, M, E - a.d3x, O - a.d3y, Ia, Sa, aa, ua, V, fa, t, r, ba, $a, kb, oa), x.columnWidth = Math.abs(ha), x.columnHeight = Math.abs(M))

} else {

"3d" == I ? (U = G - (u / 2 - this.depthCount + 1) \* (k + R) + R / 2 + E \* ca, Q = L + O \* ca, va = ca) : (U = G - (u / 2 - ca) \* (k + R) + R / 2, Q = L, va = 0);

M = k;

ya = U + k / 2;

za = F;

isNaN(L) || L < F && !x.isNegative && !f.reversed && (za = L);

ia = U + k / 2;

ja = F;

U + M > n + va \* E && (M = n - U + va \* E);

U < va \* E && (M += U - va \* E, U = va \* E);

ha = F - L;

var bc = Q;

Q = d.fitToBounds(Q, this.dy, p);

ha += bc - Q;

ha = d.fitToBounds(ha, -Q + O \* ca, p - Q);

U < n + ca \* E && 0 < M && (this.showOnAxis && (Q -= O / 2), pa = new d.Cuboid(h, M, ha, E - a.d3x, O - a.d3y, Ia, Sa, aa, ua, this.lineAlpha, fa, t, r, ba, $a, kb, oa), x.columnHeight = Math.abs(ha), x.columnWidth = Math.abs(M))

}

}

if (pa && (xa = pa.set, d.setCN(a, pa.set, "graph-" + this.type), d.setCN(a, pa.set, "graph-" + this.id), x.className && d.setCN(a, pa.set, x.className, !0), x.columnGraphics = xa, xa.translate(U, Q), this.columnsSet.push(xa), (x.url || this.showHandOnHover) && xa.setAttr("cursor", "pointer"), !P)) {

"none" == I &&

(J = r ? (this.end + 1 - K) \* D - c : D \* K + c);

"3d" == I && (r ? (J = (this.end + 1 - K) \* D - c - 1E3 \* this.depthCount, ya += E \* this.columnIndex, ia += E \* this.columnIndex, x.y += E \* this.columnIndex) : (J = (D - c) \* (K + 1) + 1E3 \* this.depthCount, za += O \* this.columnIndex, ja += O \* this.columnIndex, x.y += O \* this.columnIndex));

if ("regular" == I || "100%" == I) J = r ? 0 < Ca.value ? (this.end + 1 - K) \* D + c : (this.end + 1 - K) \* D - c : 0 < Ca.value ? D \* K + c : D \* K - c;

this.columnsArray.push({column: pa, depth: J});

x.x = r ? Q + M / 2 : U + M / 2;

this.ownColumns.push(pa);

this.animateColumns(pa, K, G, N, F, L);

this.addListeners(xa,

x)

}

}

break;

case "candlestick":

if (!isNaN(na) && !isNaN(Y)) {

var Ua, ab;

ua = Ba;

void 0 != x.lineColor && (ua = x.lineColor);

if (r) {

if (Q = F - k / 2, U = N, M = k, Q + M > p && (M = p - Q), 0 > Q && (M += Q, Q = 0), Q < p && 0 < M) {

var zb, Ab;

Y > na ? (zb = [G, Pa], Ab = [N, Oa]) : (zb = [N, Pa], Ab = [G, Oa]);

!isNaN(Pa) && !isNaN(Oa) && F < p && 0 < F && (Ua = d.line(h, zb, [F, F], ua, V, aa), ab = d.line(h, Ab, [F, F], ua, V, aa));

ha = G - N;

pa = new d.Cuboid(h, ha, M, E, O, Ia, Wa, aa, ua, V, fa, t, r, ba, $a, kb, oa)

}

} else if (U = G - k / 2, Q = L + aa / 2, M = k, U + M > n && (M = n - U), 0 > U && (M += U, U = 0), ha = F - L, U < n && 0 < M) {

xb && Y >= na && (Sa = 0);

var pa =

new d.Cuboid(h, M, ha, E, O, Ia, Sa, aa, ua, V,