Analysis of 77084 zip code using Facebook Prophet

Imports and loading csv

In [1]:
```
from fbprophet import Prophet
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.metrics import mean_squared_error, r2_score, mean_absolute_error
import warnings
warnings.filterwarnings('ignore')
#Supress default INFO logging
import logging
logger = logging.getLogger()
logger.setLevel(logging.CRITICAL)
import logging, sys
logging.disable(sys.maxsize)
from fbprophet.diagnostics import cross_validation
```

In [2]:
```
df=pd.read_csv('df_zillow_77084_prepped_fbprophet.csv')
```

In [3]:
```
df.head()
```

Out[3]:

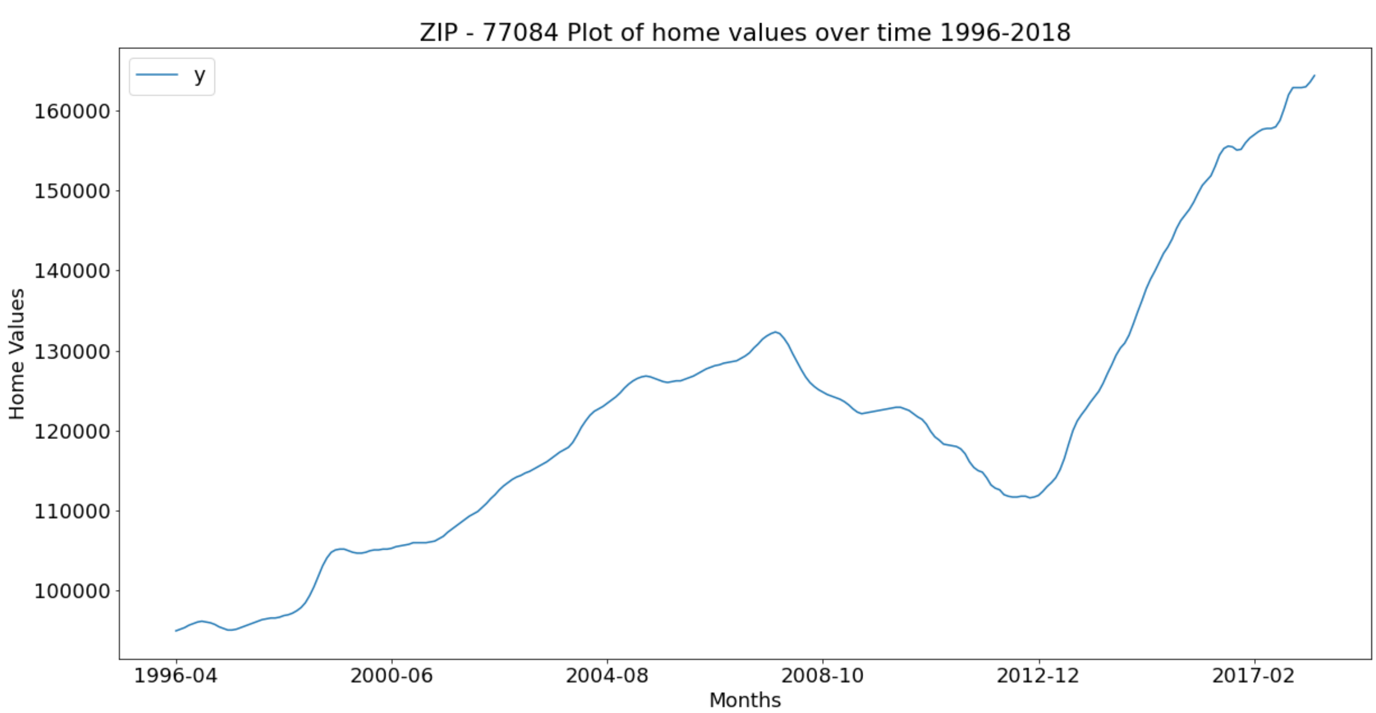
	ds	y
0	1996-04	95000.0
1	1996-05	95200.0
2	1996-06	95400.0
3	1996-07	95700.0
4	1996-08	95900.0

Plotting the specific zip code data from csv

In [15]:
```
plt.figure()
plt.rcParams.update({'font.size': 18})
ax = df.plot(title='ZIP - 77084 Plot of home values over time 1996-2018', figsize=(20,10),
ax.set_xlabel('Months')
ax.set_ylabel('Home Values')
```

Out[15]:
```
Text(0, 0.5, 'Home Values')

<Figure size 432x288 with 0 Axes>
```



Fitting and forecasting the model

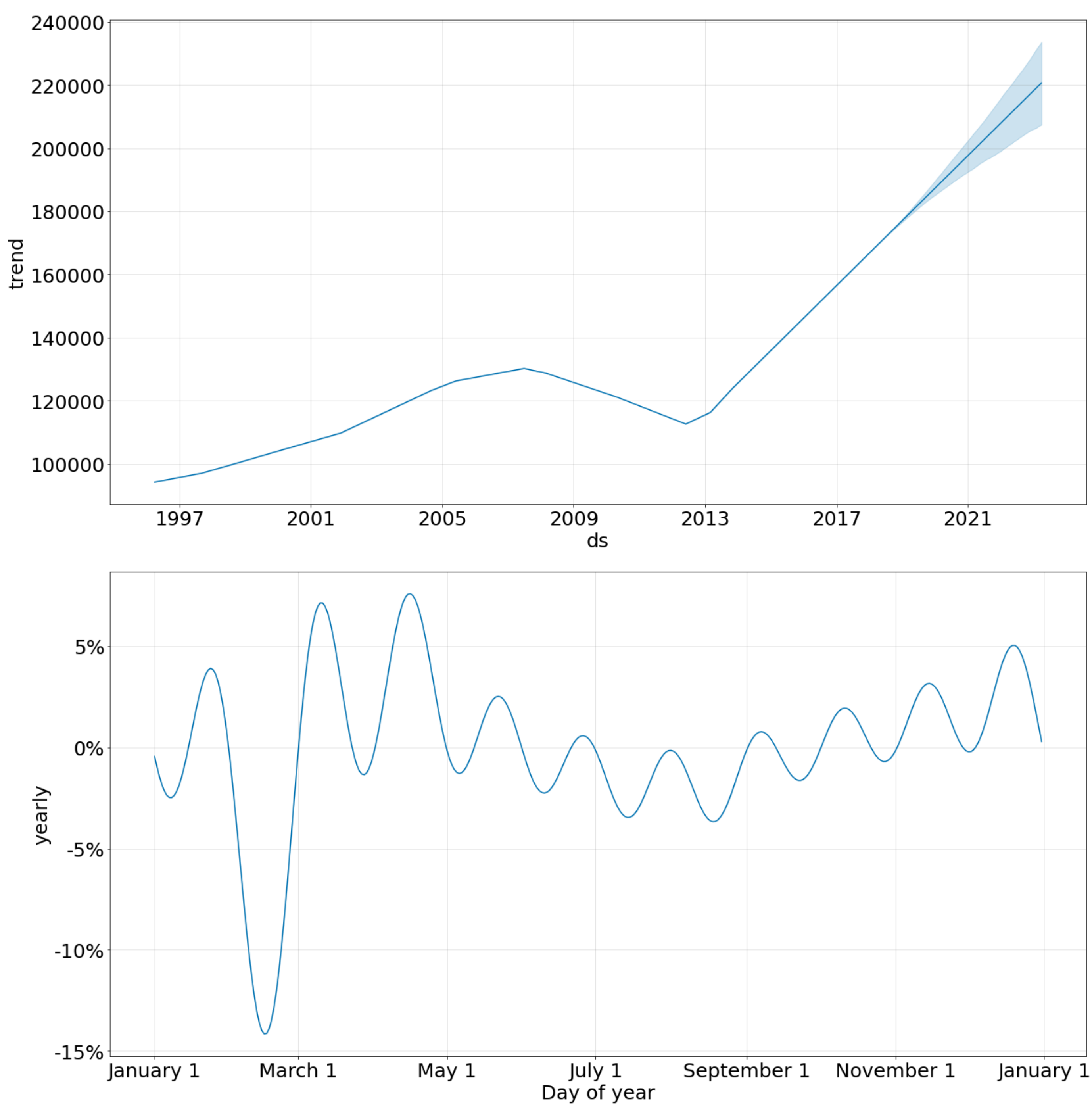
1. The length of the forecast will be 5 years into the future.
2. Periods = 60 with freq = M
    - 60 months / 12 months per year = 5 years
3. seasonality_mode = multiplicative
    - This is because additive would mean our graph will have a STEADY upward climb
    - This is not the case. There is a HUGE upward climb. Thus, multiplicative was used instead of additive.

Observations:

1. The trend shows promise, reflects the growth and demand of properties in the area.
2. The 2008 crash is reflected in the dip in home prices. This should not be confused for a cyclical occurence.
3. We cannot say much about seasonality. There is a huge upward trend.
    - Future work - maybe find stronger seasonality in daily data instead of monthly.

In [5]:
```python
m = Prophet(seasonality_mode='multiplicative').fit(df)
future = m.make_future_dataframe(periods=60, freq='M')
fcst = m.predict(future)
plt.figure()
plt.rcParams.update({'font.size': 25})
fig = m.plot_components(fcst, figsize=(20,20))
```

<Figure size 432x288 with 0 Axes>



In [6]:
```
fcst[['ds', 'yhat', 'yhat_lower', 'yhat_upper']].tail()
```

Out[6]:

	ds	yhat	yhat_lower	yhat_upper
320	2022-11-30	217034.709600	204864.277676	228711.796498
321	2022-12-31	219161.873191	206592.560330	231206.406944
322	2023-01-31	220983.592396	207837.506245	233948.932625
323	2023-02-28	214899.557381	202288.826811	227546.616941
324	2023-03-31	218788.728888	205300.098029	231663.942035

Forecast Model Diagnostics

Here I will check the accuracy of the model using cross validation

Cross validation parameters are as follows:

```
1. Model will be "m" from above fitted by Prophet() method
2. The initial training lengh parameter will be 5475 days or 15 years (365*15 = 54
75)
     - This means cutoff will be after 15 years (1996 - 2011)
3. The horizon will be 1825 days or 5 years (365 * 5 = 1825)
     - from 2012 - 2017
4. The period is set to 180 days
     - Means it will make a prediction roughly every 6 months
```

In [7]:
```
cv_results = cross_validation( model = m, initial = pd.to_timedelta(5475, unit="d"),period
```

A Jupyter widget could not be displayed because the widget state could not be found. This could happen if the kernel storing the widget is no longer available, or if the widget state was not saved in the notebook. You may be able to create the widget by running the appropriate cells.

In [8]:
```
cv_results.head()
```

Out[8]:

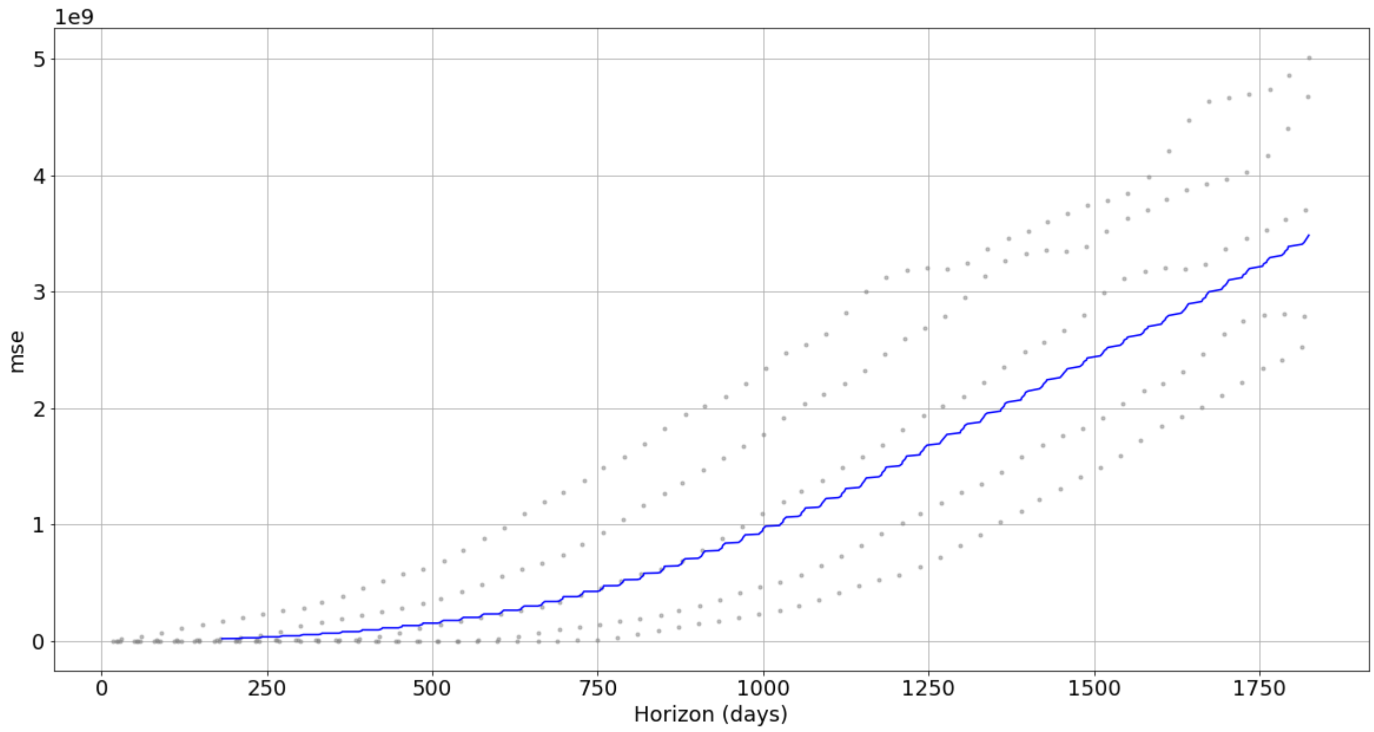
	ds	yhat	yhat_lower	yhat_upper	y	cutoff
0	2011-05-01	117948.222349	116820.577308	119143.756905	118000.0	2011-04-13
1	2011-06-01	117649.533933	116485.975168	118734.432598	117700.0	2011-04-13
2	2011-07-01	117319.662262	116159.617033	118447.369368	117100.0	2011-04-13
3	2011-08-01	116948.712952	115775.149739	118141.317740	116100.0	2011-04-13
4	2011-09-01	116591.253748	115421.764822	117750.776126	115400.0	2011-04-13

MSE observation:

```
1. MSE starts to increase exponentially after 750-1000 days.
2. This reflects higher uncertainty the farther into the horizon
```

In [9]:
```
from fbprophet.plot import plot_cross_validation_metric
plt.figure()
plt.rcParams.update({'font.size': 18})
fig = plot_cross_validation_metric(cv_results, metric='mse', figsize=(20,10))
```

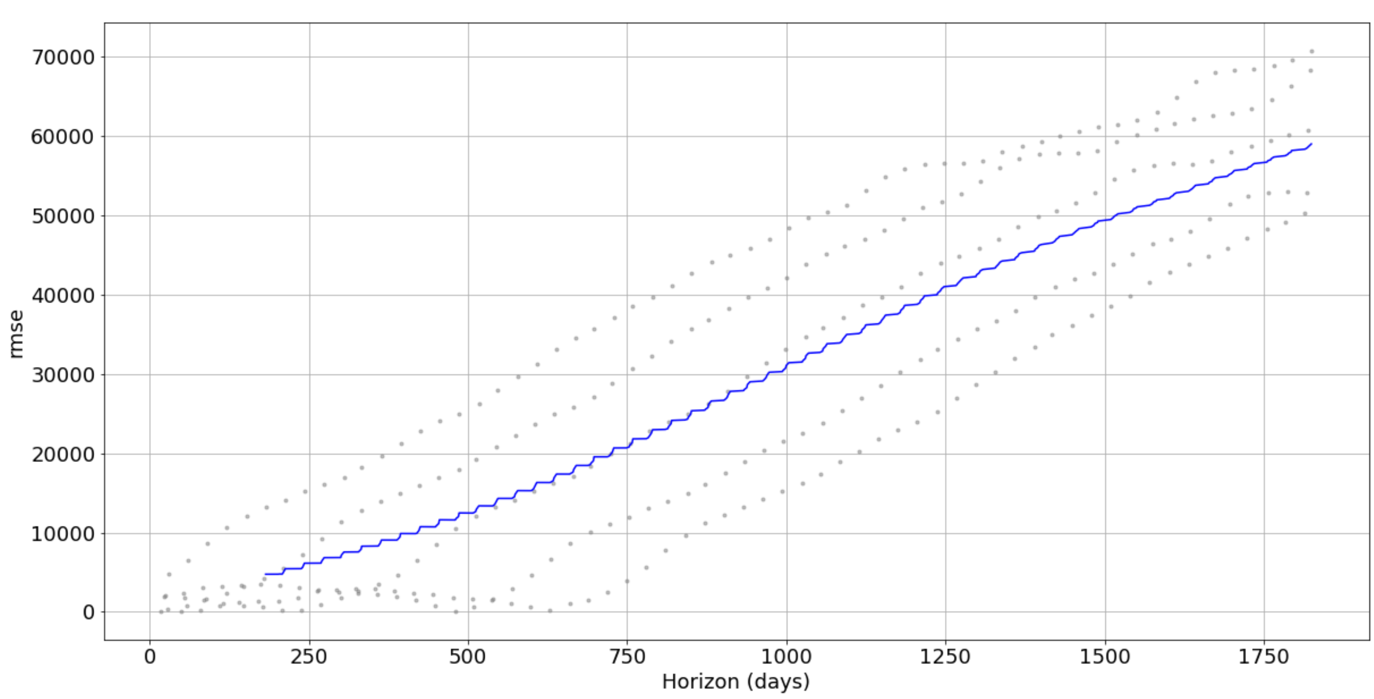
<Figure size 432x288 with 0 Axes>



RMSE Observation:

1. Similar to MSE, the error increases with longer time into horizon

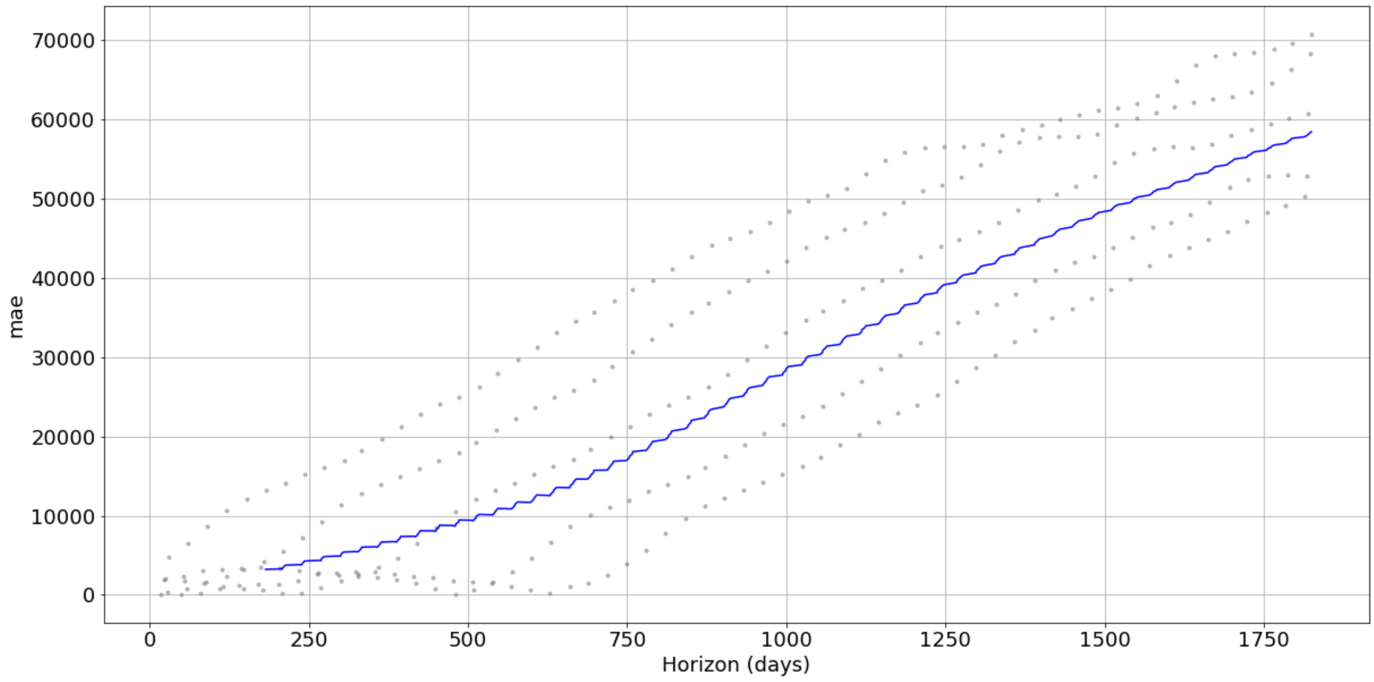
In [10]:
```
fig = plot_cross_validation_metric(cv_results, metric='rmse', figsize=(20,10))
```



MAE - Mean Absolute Error Observation:

1. Similar to MSE and RMSE, the difference starts increasing as the number of obse
rvations are increasing

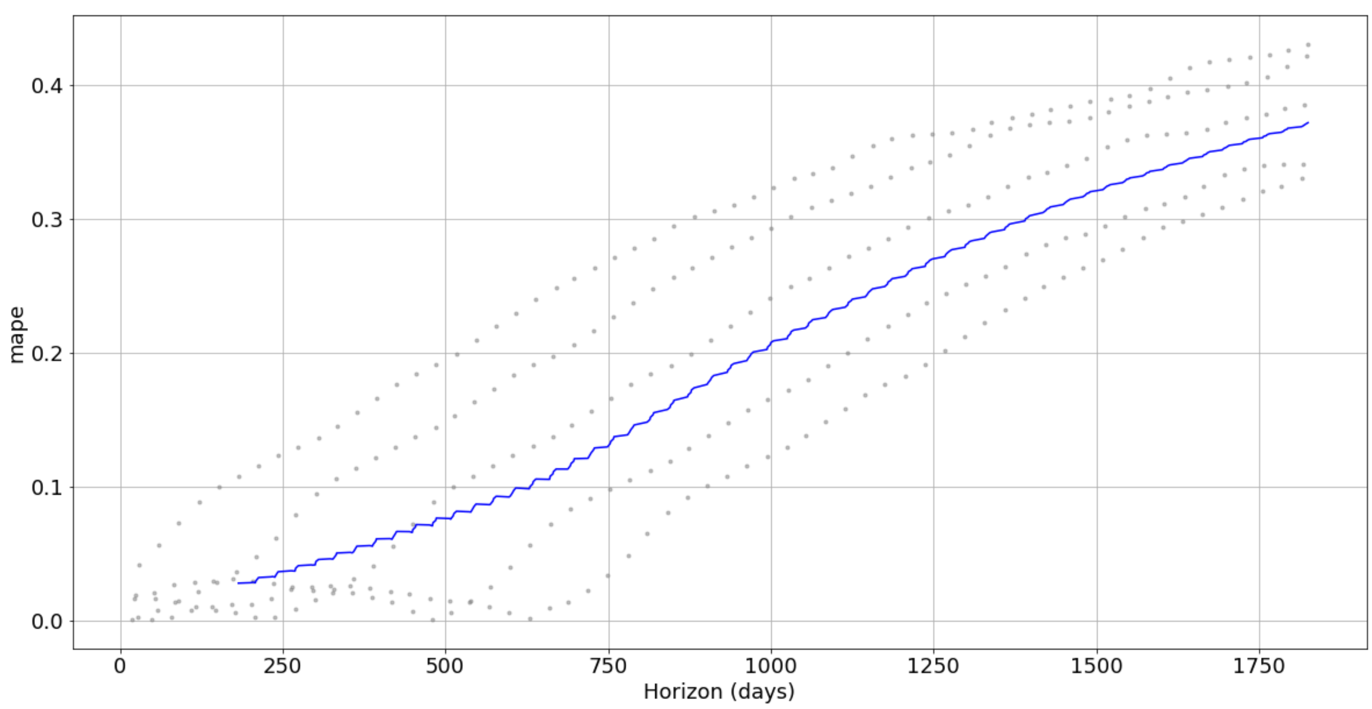
In [11]:
```
fig = plot_cross_validation_metric(cv_results, metric='mae', figsize=(20,10))
```



MAPE (Mean Average Percent Error) - Observation:

```
1. We see that MAPE increases over time
2. I am willing to tolerate MAPE of 0.1 to 0.2
    - This gets exceeded after about 1100 days
3. We will focus on MAPE as our main diagnostic metric.
    - Shows the model was about 80% accurate at 1000 days
    - Bullish prediction for the next 2-3 years
    - Supports the high upward trend we saw in the graph of all the data points fo
r the zip code
```

In [12]:
```
fig = plot_cross_validation_metric(cv_results, metric='mape', figsize=(20,10))
```



Coverage Observation:

```
1. We see coverage decreasing over time of horizon
2. Shows 0 probability as we go past about 800 days.
3. Means after about 800 days there is no probability a true value will be in the
 predicted range.
4. This does not fully explain anything. Just shows the values are not in the pred
icted range after some time.
```

In [13]:
```
fig = plot_cross_validation_metric(cv_results, metric='coverage', figsize=(20,10))
```